MXN5-C

Command Strings

Shure MXN5-C command strings for third-party control systems, such as AMX, Crestron, or Extron. Includes all supported programming commands.
Version: 1.2 (2022-K)

Table of Contents

MXN5-C Command Strings

Using a Third-Party Control System

This device can be controlled using a third-party control system with the appropriate command string.

Common applications:

- Mute
- LED color and behavior
- Loading presets
- Adjusting levels

The device is connected via Ethernet to a control system, such as AMX, Crestron or Extron.

- **Connection:** Ethernet (TCP/IP; select “Client” in the AMX/Crestron program)
- **Port:** 2202

If using static IP addresses, set the Shure Control and the Audio Network settings to Manual in Designer. Use the Control IP address for TCP/IP communication with Shure devices.

See below for all supported command strings. This list is updated with each firmware release.

Channel Number Assignments

MXN5-C uses the following numbering to distinguish the channels for REP values. The channels use 2 digits even if the channel number is less than 10.

- Dante inputs: 01-02
- Summed input (no Dante name): 03
- Dante output: 04

Command String Conventions

When you make changes to a parameter, the device sends a REPORT string with information about what you changed. You don't need to constantly query parameters.

All messages are ASCII, including level and gain indicators.

This device uses 4 types of strings:

- **GET**
  - Finds the status of a parameter. The device responds with a REPORT string.
- **SET**
  - Changes the status of a parameter. The device responds with a REPORT string that shows the parameter's new value.
- **REP**
  - The device sends REPORT strings to show the status of parameters anytime a parameter changes.

- **SAMPLE**
  - Used for metering audio levels.

Get All

Parameter Name:	ALL
Command Types Supported:	GET, REP
Indexing:	n/a
Value(s):	Responds with `REP` for all device-specific properties and `ALL` channel-related properties.
Example(s):	< GET ALL >

Model

Parameter Name:	MODEL
Command Types Supported:	GET, REP
Indexing:	n/a
Value(s):	**model** is a 32 character quoted string. The value is padded with spaces to ensure that 32 characters are reported.
Example(s):	< GET MODEL > < REP MODEL model >

Serial Number

Parameter Name:	SERIAL_NUM
Command Types Supported:	GET, REP
Indexing:	n/a
Value(s):	**serial_num** is a 32 alphanumeric character string. Response is padded to ensure that 32 characters are always returned
Example(s):	< GET SERIAL_NUM > < REP SERIAL_NUM serial_num >

Firmware Version

Parameter Name:	FW_VER
Command Types Supported:	GET, REP
Indexing:	n/a
Value(s):	Where **ver** is an 18 character literal string: The value is 3 versions separated by a period. Each version shall be able to take on a value from 0 to 65535. **ver** has an "*" if the firmware is invalid. **Example**: 65535.65535.65535
Example(s):	< GET FW_VER > < REP FW_VER ver >

IP Address for Primary Audio Network

Parameter Name:	IP_ADDR_NET_AUDIO_PRIMARY
Command Types Supported:	GET, REP
Indexing:	n/a
Value(s):	The value of IP address consist of 4 octets each separated by a period. The length of IP address is 15 characters. The value will be padded to ensure that 15 characters are always returned.
Example(s):	< GET IP_ADDR_NET_AUDIO_PRIMARY > < REP IP_ADDR_NET_AUDIO_PRIMARY ip_addr > < REP ERR >

Subnet Mask for the Primary Audio Network

Parameter Name:	IP_SUBNET_NET_AUDIO_PRIMARY

Command Types Supported:	GET, REP
Indexing:	n/a
Value(s):	**subnet** is subnet mask: 32 bit number represented in the Binary Coded Decimal notation in the form of A.B.C.D where each variable A or B or C or D are 8 bit octets each separated by a period. The length of subnet is 15 characters. The value will be padded to ensure that 15 characters are always returned.
Example(s):	< GET IP_SUBNET_NET_AUDIO_PRIMARY > < REP IP_SUBNET_NET_AUDIO_PRIMARY subnet > < REP ERR >

Network Gateway for Primary Audio Network Interface

Parameter Name:	IP_GATEWAY_NET_AUDIO_PRIMARY
Command Types Supported:	GET, REP
Indexing:	n/a
Value(s):	**gateway** is network gateway: 32 bit number represented in the Binary Coded Decimal notation in the form of A.B.C.D where each variable A or B or C or D are 8 bit octets each separated by a period. The length of subnet is 15 characters. The value will be padded to ensure that 15 characters are always returned.
Example(s):	< GET IP_GATEWAY_NET_AUDIO_PRIMARY > < REP IP_GATEWAY_NET_AUDIO_PRIMARY gateway > < REP ERR >

Control MAC Address

Parameter Name:	CONTROL_MAC_ADDR
Command Types Supported:	GET, REP
Indexing:	n/a

Value(s):	**addr** is a 17 character literal string formatted as 6 octets, each separated by a colon. **Example:** 00:0E:DD:FF:F1:63
Example(s):	< GET CONTROL_MAC_ADDR > < REP CONTROL_MAC_ADDR addr > < REP ERR >

Device ID

Parameter Name:	DEVICE_ID
Command Types Supported:	GET, REP
Indexing:	n/a
Value(s):	Response is a text string. Most devices allow device ID to be up to 31 characters. Value is padded with spaces as needed to ensure that 31 characters are always reported
Example(s):	< GET DEVICE_ID > < REP DEVICE_ID string >

Network Audio (Dante) Device Name

Parameter Name:	NA_DEVICE_NAME
Command Types Supported:	GET, REP
Indexing:	n/a
Value(s):	Response is a text string. Most devices allow device ID to be up to 31 characters. Value is padded with spaces to ensure that 31 characters are always reported.
Example(s):	< GET NA_DEVICE_NAME > < REP NA_DEVICE_NAME string >

Channel Name

Parameter Name:	CHAN_NAME
Command Types Supported:	GET, REP
Indexing:	GET **index:** See Channel Number Assignment for product-specific channel assignments. 0 = all channels. REP **index:** 2 digit representation of the index sent in the GET, all the appropriate channels if the **index** = 0.
Value(s):	**string** is 31 character channel name. Value is padded with spaces as needed to ensure that 31 characters are always reported.
Example(s):	< GET index CHAN_NAME > < REP index CHAN_NAME string > < REP ERR >

Network Audio (Dante) Channel Name

Parameter Name:	NA_CHAN_NAME
Command Types Supported:	GET, REP
Indexing:	GET **index:** See Channel Number Assignment for product-specific channel assignments. 0 = all channels. REP **index:** 2 digit representation of the index sent in the GET, all the appropriate channels if the **index** = 0.
Value(s):	**string** is 31 character channel name. Value is padded with spaces as needed to ensure that 31 characters are always reported.
Example(s):	< GET index NA_CHAN_NAME > < REP index NA_CHAN_NAME string > < REP ERR >

Identify Device (Flash LED)

Parameter Name:	FLASH

Command Types Supported:	GET, SET, REP
Indexing:	n/a
Value(s):	**flash_state** takes on values ON OFF
Example(s):	< GET FLASH > < SET FLASH flash_state > < REP FLASH flash_state > < REP ERR >

Audio Clip Indicator

Parameter Name:	AUDIO_OUT_CLIP_INDICATOR
Command Types Supported:	GET, REP
Indexing:	GET **index** : See Channel Number Assignment for product-specific channel assignments. 0 = all channels. REP **index** : 2 digit representation of the index sent in the GET, all the appropriate channels if the **index** = 0.
Value(s):	**sts** is current status for the channel: 1. OFF 2. ON
Example(s):	< GET index AUDIO_OUT_CLIP_INDICATOR > < REP index AUDIO_OUT_CLIP_INDICATOR sts > < REP ERR >

Metering Rate (RMS)

Parameter Name:	METER_RATE
Command Types Supported:	GET, SET, REP
Indexing:	n/a

Value(s):	**rate** is a value from 100 to 99999 representing meter rate in milliseconds. 0 = off Values 1 to 99 are not valid and result in response. **aaa bbb ccc ddd -** Audio Levels take on values 000-060, which represent actual audio levels of -60 to 0 dBFS. Represent channels in order defined in Channel Number Assignment.
Example(s):	< GET METER_RATE > < SET METER_RATE rate > < REP METER_RATE rate > < REP ERR > < SAMPLE aaa bbb ccc ddd >

Audio Gain (Digital)

Parameter Name:	AUDIO_GAIN_HI_RES
Command Types Supported:	GET, SET (INC, DEC), REP
Indexing:	GET **index:** See Channel Number Assignment for product-specific channel assignments. 0 = all channels. REP **index:** 2 digit representation of the index sent in the GET, all the appropriate channels if the **index** = 0. Setting gain on all channels at once is not supported.
Value(s):	**gain** is in units of one-tenth of a dB. The value is multiplied by 10 and then scaled by 1100. The resulting value has a range of 0 to 1400 representing gain from -110.0 dB to 30.0 dB. **step** is in units of one-tenth of a dB. The resulting gain when the **step** is applied must be in the range allowed in the SET.
Example(s):	< GET index AUDIO_GAIN_HI_RES > < SET index AUDIO_GAIN_HI_RES gain > < SET index AUDIO_GAIN_HI_RES inc step > < SET index AUDIO_GAIN_HI_RES dec step > < REP index AUDIO_GAIN_HI_RES gain > < REP ERR >

Device Mute

Parameter Name:	DEVICE_AUDIO_MUTE
Command Types Supported:	GET, SET, REP

Indexing:	n/a
Value(s):	**cmd** is desired mute status and takes on values: ON OFF TOGGLE **sts** is the current mute status for the designated channel and takes on values: ON OFF
Example(s):	< GET DEVICE_AUDIO_MUTE > < SET DEVICE_AUDIO_MUTE cmd > < REP DEVICE_AUDIO_MUTE sts >

Channel Mute

Parameter Name:	AUDIO_MUTE
Command Types Supported:	GET, SET, REP
Indexing:	Where **nn** is the channel and takes on values defined in channel number assignments.
Value(s):	**cmd** is desired mute status and takes on values: ON OFF TOGGLE **sts** is the current mute status for the designated channel and takes on values: ON OFF
Example(s):	< GET nn AUDIO_MUTE > < SET nn AUDIO_MUTE cmd > < REP nn AUDIO_MUTE sts >

Presets

Parameter Name:	PRESET

Command Types Supported:	GET, SET, REP
Indexing:	## is the preset number and takes on values 1-10.
Value(s):	n/a
Example(s):	< GET PRESET > < SET PRESET ## > < REP PRESET ## > < REP ERR >

Restore Default Settings

Parameter Name:	DEFAULT_SETTINGS
Command Types Supported:	SET, REP
Indexing:	n/a
Value(s):	## = 00 if restore is successful
Example(s):	< SET DEFAULT_SETTINGS > < REP DEFAULT_SETTINGS ## > < REP ERR >

View Preset Name

Parameter Name:	PRESET_NAME
Command Types Supported:	GET, REP
Indexing:	1-10: specific preset identifier
Value(s):	**name** is a literal string 25 alphanumeric characters long, special characters allowed except blank spaces, {} and < >. Note that if a preset is empty, **name** will say {empty}
Example(s):	< GET PRESET_NAME nn > < REP PRESET_NAME nn name > < REP ERR >

Limiter Engaged

Parameter Name:	LIMITER_ENGAGED
Command Types Supported:	GET, REP
Indexing:	# = Channel 3
Value(s):	**sts** indicates whether the limiter is engaged or not and takes on values: ON OFF
Example(s):	< GET # LIMITER_ENGAGED > < REP # LIMITER_ENGAGED sts > < REP ERR >

Device Encryption Status

Parameter Name:	ENCRYPTION
Command Types Supported:	GET, REP
Indexing:	n/a
Value(s):	**sts** is the encryption status, which can have the following values: ON OFF
Example(s):	< GET ENCRYPTION > < REP ENCRYPTION sts > < REP ERR >

Reboot

Note: This command does not send acknowledgement.

Parameter Name:	REBOOT
Command Types Supported:	SET
Indexing:	n/a
Value(s):	n/a

Example(s):	< SET REBOOT >

Get Error Events

Parameter Name:	LAST_ERROR_EVENT
Command Types Supported:	GET, REP
Indexing:	n/a
Value(s):	Sends the last error logged on the device, as represented by **{str}. {str}** is up to 128 characters long.
Example(s):	< GET LAST_ERROR_EVENT > < REP LAST_ERROR_EVENT {str} > < REP ERR >

PEQ Filter Enable

Parameter Name:	PEQ
Command Types Supported:	GET, SET, REP
Indexing:	GET **index** : See Channel Number Assignment for product-specific channel assignments. 0 = all channels. REP **index** : 2 digit representation of the index sent in the GET, all the appropriate channels if the **index** = 0. MXN5-C **index**: 03 **filter** is the filter number in the selected PEQ block **index.** 0: all filters.
Value(s):	**sts** is the desired PEQ filter status: ON OFF TOGGLE
Example(s):	< GET index PEQ filter > < SET index PEQ filter sts > < REP index PEQ filter sts > < REP ERR >

Delay

Parameter Name:	DELAY
Command Types Supported:	GET, SET, REP
Indexing:	**index** is selected output channels that have delay feature. Channels are defined in Channel Number Assignment. GET **index** : Selected output channels that support delay. 0 = all relevant channels. MXN5-C: Channel 03 REP **index:** Double-digit representation of the index sent in the GET, all the appropriate channels if the **index** = 0.
Value(s):	#### is delay data in 1 ms increment. Delay range: 0 means delay block is disabled. MXN5-C: 1 to 160 ms
Example(s):	< GET index DELAY > < SET index DELAY #### > < REP index DELAY #### > < REP ERR >

Bypass DSP

Allows you to bypass or enable these DSP blocks: EQ, delay, and limiter.

Parameter Name:	BYPASS_DSP
Command Types Supported:	GET, SET, REP
Indexing:	n/a
Value(s):	**sts** takes on values: ON OFF TOGGLE
Example(s):	< GET BYPASS_DSP > < SET BYPASS_DSP sts > < REP BYPASS_DSP sts > < REP ERR >

Signal Generator Type

Allows you to set and view the signal generator type.

Parameter Name:	SIG_GEN_TYPE
Command Types Supported:	GET, SET, REP
Indexing:	GET **index:** 0 or 3 REP **index:** 2 digit representation of the index sent in the GET, all the appropriate channels if the **index** = 0
Value(s):	**type** is type of signal generator: PINK WHITE TONE SWEEP
Example(s):	< GET index SIG_GEN_TYPE > < SET index SIG_GEN_TYPE type > < REP index SIG_GEN_TYPE type > < REP ERR >

Signal Generator Tone Frequency

Allows you to set and view the signal generator frequency.

Parameter Name:	SIG_GEN_FREQ
Command Types Supported:	GET, SET, REP
Indexing:	GET **index:** 0 or 3 REP **index:** 2 digit representation of the index sent in the GET, all the appropriate channels if the **index** = 0
Value(s):	**sts** is a single frequency in the range of 125 to 20,000 Hz, in 1 Hz increments.
Example(s):	< GET index SIG_GEN_FREQ > < SET index SIG_GEN_FREQ sts > < REP index SIG_GEN_FREQ sts > < REP ERR >

Signal Generator Gain

Parameter Name:	SIG_GEN_GAIN
Command Types Supported:	GET, SET, REP
Indexing:	GET **index:** 0 or 3 REP **index:** 2 digit representation of the index sent in the GET, all the appropriate channels if the **index** = 0
Value(s):	**gain** is in the range of 0-1310, which represents -110.0 to 21.0 dB in 0.1 dB increment. The dB value is first converted to integer and then scaled by 1100.
Example(s):	< GET index SIG_GEN_GAIN > < SET index SIG_GEN_GAIN gain > < REP index SIG_GEN_GAIN gain > < REP ERR >

Signal Generator Start/Stop

Parameter Name:	SIG_GEN
Command Types Supported:	SET, REP
Indexing:	GET **index:** 0 or 3 REP **index:** 2 digit representation of the index sent in the GET, all the appropriate channels if the **index** = 0
Value(s):	**sts** can be: START STOP TOGGLE
Example(s):	< SET index SIG_GEN sts > < REP index SIG_GEN sts > < REP ERR >

Contact Customer Support

Didn't find what you need? Contact our customer support to get help.