



IntelliMix Room

Command Strings

Shure IntelliMix Room command strings for third-party control systems, such as AMX, Crestron, or Extron. Includes all supported programming commands.

Version: 4 (2020-C)

Table of Contents

IntelliMix RoomCommand Strings	3	Using Command Strings with IntelliMix® Room	3
---------------------------------------	----------	--	----------

IntelliMix Room Command Strings

Using Command Strings with IntelliMix[®] Room

To control IntelliMix Room with third-party control systems, turn on command strings in Designer. You must choose an open port on the computer running IntelliMix Room to send and receive command strings.

Default port: 2202

Port range: 1025-65534

1. In Designer, open the installation of IntelliMix Room you want to control and go to **Settings > Command strings**.
2. Turn on command strings.
3. Enter an available port to send and receive the command strings. Make sure that the port isn't being used by any other programs installed on the computer.

To test if the port is available:

- Send a command to IntelliMix Room. Make sure you are targeting IntelliMix Room's control IP address.
- If you don't receive a response, the port is being used by another program.
- Try a different port number and send the command until you receive a response.
- For best results, install all software on the computer before choosing a port for command strings. Adding new software after setting up command strings can change how the computer's ports are used, which can affect command strings.

A complete list of command strings is available at pubs.shure.com/command-strings/IntelliMixRoom.

Channel Number Assignments

IntelliMix Room 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.

- All Channels: 00
- Dante Mic Inputs: 01-16
 - There are 8 and 16 channel versions of IntelliMix Room. If you have a version that is not the full 16 channels, command strings that refer to the unlicensed channels will fail.
- Dante Line Inputs: 17-24
- Virtual Audio Input: 25
- PC Input: 26
- Dante Outputs: 27-34
- Virtual Audio Output: 35
- PC Output: 36
- Automix Output: 37

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 >

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 >

--	--

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 >

NA 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 >

Audio Gain (Digital)

Parameter Name:	AUDIO_GAIN_HI_RES
Command Types Supported:	GET, SET (inc, dec), REP
Indexing:	Where ## is channel and takes on values defined in channel number assignments.

Value(s):	<ol style="list-style-type: none"> 1. Applies to PRE_GATE gain on devices that support both pre- and post-gain control. 2. Allows user to view and set the property. 3. Applies to inputs and mixed outputs only. 4. Requesting this parameter will result in < REP ERR > if the channel is not valid. 5. Setting gains on all channels at once is not supported. <p>## is the channel defined in the GET command.</p> <p>gain is in units of one-tenth of a dB and then scaled by 1100 and takes on values 0 to 1400 representing gain from -109.9 dB to 30.0 dB.</p> <p>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.</p>
Example(s):	<pre>< GET ## AUDIO_GAIN_HI_RES > < SET ## AUDIO_GAIN_HI_RES gain > < SET ## AUDIO_GAIN_HI_RES inc step > < SET ## AUDIO_GAIN_HI_RES dec step > < REP ## AUDIO_GAIN_HI_RES gain > < REP ERR ></pre>

Device Mute

Parameter Name:	DEVICE_AUDIO_MUTE
Command Types Supported:	GET, SET, REP
Indexing:	n/a
Value(s):	<p>cmd is desired mute status and takes on values:</p> <p>ON OFF TOGGLE</p> <p>sts is the current mute status for the designated channel and takes on values:</p> <p>ON OFF</p>
Example(s):	<pre>< GET DEVICE_AUDIO_MUTE > < SET DEVICE_AUDIO_MUTE cmd > < REP DEVICE_AUDIO_MUTE sts ></pre>

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):	<p>cmd is desired mute status and takes on values:</p> <p>On Off Toggle</p> <p>sts is the current mute status for the designated channel and takes on values:</p> <p>ON OFF</p>
Example(s):	<pre>< GET nn AUDIO_MUTE > < SET nn AUDIO_MUTE cmd > < REP nn AUDIO_MUTE sts ></pre>

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):	<pre>< GET PRESET > < SET PRESET ## > < REP PRESET ## > < REP ERR ></pre>

Automixer Post-Gate Mute

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

Indexing:	xx is 01-16 for input channels and 37 is for the automix output.
Value(s):	<p>cmd is desired mute status and takes on values:</p> <p>On Off Toggle</p> <p>sts is the current mute status for the designated channel and takes on values:</p> <p>ON OFF</p>
Example(s):	<pre>< GET xx AUTOMXR_MUTE > < SET xx AUTOMXR_MUTE cmd > < REP xx AUTOMXR_MUTE sts > < REP ERR ></pre>

Audio Gain Postgate

Parameter Name:	AUDIO_GAIN_POSTGATE
Command Types Supported:	GET, SET, REP
Indexing:	xx is 01-16 for input channels and 37 is for the automix output.
Value(s):	<p>gain is in units of one-tenth of a dB and then scaled by 1100 and takes on values 0 to 1400 representing gain from -109.9 dB to 30.0 dB</p> <p>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.</p>
Example(s):	<pre>< GET xx AUDIO_GAIN_POSTGATE > < SET xx AUDIO_GAIN_POSTGATE gain > < SET xx AUDIO_GAIN_POSTGATE inc step > < SET xx AUDIO_GAIN_POSTGATE dec step > < REP xx AUDIO_GAIN_POSTGATE gain > < REP ERR ></pre>

Automixer Gate Status

Parameter Name:	AUTOMXR_GATE
Command Types Supported:	GET, REP

Indexing:	xx is the channel number (see channel number assignments).
Value(s):	sts is the current mute status for the designated channel and takes on values: ON OFF
Example(s):	< GET xx AUTOMXR_GATE > < REP xx AUTOMXR_GATE sts > < REP ERR >