AC-MXNET-CBOX, AC-MXNET-CBOX-B, AC-MXNET-CBOX-HA API Command List

Version 1.21.1

The CBOX's API server uses TCP port 24. API Commands can also be sent from the "System Log" page of the Mentor web GUI for testing/troubleshooting purposes.

NOTE: Certain command feedback responses may appear to have a newline within this document. This is NOT the case for the actual feedback response, this is how the text is formatted within the document.
NOTE: All commands, unless specifically stated otherwise, do not use any brackets or quotes: "" [] {} to execute a command.

Within the API "config set device" commands, the last command argument will always be the targeted encoder(s) or decoder(s). Abbreviations such as "EncID" means the Encoder's Custom Name, or "EncMAC" means the encoder's MAC address. Same goes with "DecID/DecMAC" for MXnet decoders.

1.21.1	Added the command 'config set device audiosrc [1,2] {DeviceID/DeviceMAC} Added the command 'config set device sourceinput {1,2} [EncID/EncMAC]'. Applicable for certain encoders such as the AC-MXNET-1G-EV2WP. Added the command 'config set device dspmode {1,2} [EncID/EncMAC]' Added the command 'config get device routes [v,a,u,r,s] [DecID/DecMAC]' The following commands are only available on V2 FPGA 9.18.9 or greater: Added the command 'config set device shareusb {0,1} [EncID/EncMAC/DecID/DecMAC]' Added the command 'config set device ull {0,1} [DecID/DecMAC]'
1.19.1	Reworded several API command descriptions/examples to fix typos Added Audio Return Path (ARC/eARC) Commands. V2 endpoints only!! Added the command 'config set device srckeeptime {time} [DecoderID/DecoderMAC/ALLRX]' Added the command 'config set device usb5v {0,1,2} [EncoderID/EncoderMAC] [DecoderID/DecoderMAC]' Added the command 'config set device kmoip [0,1] [DeviceID/DeviceMAC]' Added the command 'config set device audiomute [0~10000] {DecoderID/DecoderMAC}' Added the command 'config get device audiomute {DecoderID/DecoderMAC}'
1.18.1	Added the command 'config set device usbmode [0,1] {DeviceID/DeviceMAC}' Added the command 'config set previews [on,off]' Added the command 'config set device hdcprequest' Added the API commands for Matrix Preset lists. Reworded the majority of API commands to be more descriptive.

1.17	Add command 'config set device ediddata {EDID DATA} {TxID/TxMAC}'
1.16	Add command 'vwid layout tx {videowall_name} {layout_name} {row:column/OldTXID/OldTXMAC/ALL} {TXID/TXMAC}'. Add command 'config set device usbpath {TxID/TxMAC} {RxID/RxMAC}' and 'config set device usbpathdisable {RxID/RxMAC}'. Command 'config set device rs232responsetype', add option 3, HEX. Modify command 'config set device hdcp XX {RxID/RxMAC}', only support options 2=HDCP OFF, 3=HDCP1.4, 4=HDCP2.2.
1.15	Add commands 'config set device osdmode 1/2/3 {RxID/RxMAC/RxID1:RxID2:RxID3:.../RxMAC1:RxMAC2:RxMAC3:.../ALLRX/ALL}', 'config set device osdfontsize [font size] {RxID/RxMAC/RxID1:RxID2:RxID3:.../RxMAC1:RxMAC2:RxMAC3:.../ALLRX/ALL}', 'config set device osdcolor [hex color] {RxID/RxMAC/RxID1:RxID2:RxID3:.../RxMAC1:RxMAC2:RxMAC3:.../ALLRX/ALL}', 'config set device exaudio volume {VALUE} {TxID/TxMAC}', 'config set device pattern 0/1/2 {RxID/RxMAC}'.
1.14	Add commands 'config set device osdsize 1/2 {RxID/RxMAC}', 'config set device displaypicture 1/2/3/4/5 {RxID/RxMAC}', 'config set device blackout on/off {RxID/RxMAC}', 'config set device quality -1/0/1/2/3/4/5 {TxID/TxMAC}'.
1.13	Add command 'config set device rs232responsetype 1/2 {TxID/TxMAC/TxID1:TxID2:TxID3:.../TxMAC1:TxMAC2:TxMAC3:.../TxMACALLTX/ALLRX/ALL}'. Change cec api to 'config set device cec {hexData}[,{hexData},{hexData}...] {device_id/device_mac}'. Support to send one or multiple cec data to device, separated by ','. Add command 'config set device avdmid {new_id} {TxID/TxMAC}'. Add command 'config set device avdmdes {new_description} {TxID/TxMAC}'. Remove exmxmode option [0-Matrix Mode Close]. Change command description of 'config set device rs232 dataType {serial port data} {device_id/device_mac}'. Change command description of 'vw add'/'vwid add'/'vwid layout add'/'mosaic add'/'vw mosaic layout add'. Update some api description.
1.12	Add command 'config set device exmxmode x {TxID/TxMAC}'. Add command 'config set device stream on/off {RxID/RxMAC or RxMAC1:RxMAC2:...}'. Add command 'config get date'. Add command 'config set date {year} {month} {day} {hour} {minute} {second}'. Add command 'config get timezone'. Add command 'config set timezone {AAA+B/AAA-B}'. Add command 'config get ntp'. Add command 'config set ntp {ntpserverip1} [ntpserverip2] [ntpserverip3] [ntpserverip4] [ntpserverip5]'. Add command ''Remove EDID option 0. Remove config set device capture. No longer needed.
1.11	Add a series of mosaic related commands. Add command 'config set device videopath {TxID/TxMAC} {RxID/RxMAC}', 'config set device

	videopathdisable {RxID/RxMAC}', 'config set device audiopath {TxID/TxMAC} {RxID/RxMAC}', 'config set device audiopathdisable {RxID/RxMAC}'.
1.10	Command 'set device ir', 'set device cec', 'set device rs232', 'set device rs232setting' can support multiple device(**separated by colon**) or **ALL**/**ALLRX**/**ALLTX**.
1.09	Command 'config set device cec', support send cec to multiple RX's(less than 50, **separated by colon**), using MAC address or CUSTOM NAME, or ALL RX's. Command 'config set device osd', support set osd to multiple RX's(less than 50, **separated by colon**), using MAC address or CUSTOM NAME, or ALL RX's.
1.08	Add command 'vwid layout combine' and 'vwid layout split'.
1.07	Update API document.
1.06	Add kvm command. Use for kvm roaming. Add command 'config set device hpdrst {device_id or device_mac}'
1.05	Add copy command in vwid. support to set bezel gap in vwid. Add 'config set device kvmroaming RX1ID,X1,Y1:RX2ID,X2,Y2[:RX3ID,X3,Y3...] RX7ID'.
1.02	Add vwid command.
1.01	Update 'matrix aset' and 'matrix add' command description.

CBOX System Commands

These commands are used for obtaining/configuring the MXnet Control Box itself, instead of encoders/decoders.

config get name

Function

Retrieves the hostname of the MXNet Control box (the hostname will not change).

Example command & feedback response:

config get name

"{"gid":200000,"cmd":"config get name","info":"AC-MXNET-CBOX","code":0}"

config get version

Function

Retrieves the current firmware version of the MXNet Control Box.

Example

config get version

"{"cmd":"config get version","info":"2.28","code":0}"

config get ipsetting

Function

Obtain the network configuration setting for the AV NETWORK port of the Control Box (the port that monitors and controls MXNet devices). The feedback response will output in this order: [IPSetting]/[IPAddress]/[SubnetMask]/[DefaultGateway]

Example

config get ipsetting

AUTOIP (default):

{"cmd":"config get ipsetting","info":"autoip/169.254.4.198/255.255.0.0/192.168.1.1","code":0}

DHCP:

{"cmd":"config get ipsetting","info":"dhcp/192.168.10.23/255.255.255.0/192.168.10.1","code":0}

STATIC:

"{"cmd":"config get ipsetting","info":"static/192.168.99.1/255.255.255.0/192.168.99.254",code":0}"

config get ipsetting2

Function

Retrieve the IP configuration for the MENTOR/LAN port of the control box (the port used for connection to a local area network for accessing the CBOX's GUI/communicating to the CBOX via a third party control processor). The feedback response will output in this order: [IPSetting]/[IPAddress]/[SubnetMask]/[DefaultGateway]

Example command & feedback response:

config get ipsetting2

STATIC:

"{"cmd":"config get ipsetting2","info":"static/192.168.1.239/255.255.255.0/192.168.1.1","code":0}"

DHCP(default):

"{"cmd":"config get ipsetting2","info":"dhcp/192.168.3.7/255.255.252.0/192.168.1.1","code":0}"

config get devicelist

Function

Retrieve a list of all devices currently residing within the Control Box's database, whether online or offline. This command will also output the current configuration status of said endpoints, such as current subscriptions, IP/MAC addresses, current firmware version, etc.

Example command & feedback response:

config get devicelist

"{"cmd":"config get devicelist","info":{"188A6ACE87DC":{"ch":"0009","id":"188A6ACE87DC","ipmode":"autoip","rs232mode":"2","online":14147,"version":"3.13","state":"s_srv_on","mac":"188A6ACE87DC","is_host":1,"dtype":"ast152x","edid":"2","ip":"169.254.10.190"},"188A6A45C4A5":{"version":"3.13","id":"188A6A45C4A5","mac":"188A6A45C4A5","ch_p":"0000","dtype":"ast152x","ch_u":"0001","ch_v":"0002","ch":"0000","ch_a":"0002","ipmode":"autoip","rs232mode":"2","ch_r":"0001","ch_c":"0000","ch_s":"0002","online":14147,"state":"s_srv_on","ip":"169.254.8.59"},"188A6A0F4485":{"ch":"0002","id":"188A6A0F4485","ipmode":"autoip","rs232mode":"2","online":14147,"version":"3.11","state":"s_attaching","mac":"188A6A0F4485","is_host":1,"dtype":"ast152x","edid":"2","ip":"169.254.4.246"},"188A6A1887E3":{"version":"3.12","id":"188A6A1887E3","mac":"188A6A1887E3","ch_p":"0000","dtype":"ast152x","ch_u":"0001","ch_v":"0009","ch":"0000","vmode":"0","ch_a":"0009","ipmode":"autoip","rs232mode":"2","ch_r":"0001","ch_c":"0000","ch_s":"0001","online":14147,"state":"s_srv_on","ip":"169.254.7.251"}},"code":0}"

config get rs-232 alias

Function

Retrieve the current status of the RS232 port on the CBOX, whether it's "on" or "off".

Example command & feedback response:

config get rs-232 alias

"{"cmd":"config get rs-232 alias","info":"on","code":0}"

config get dns

Function

Obtain the DNS configurations for the CBOX.

Example command & feedback response:

config get dns

"{"cmd":"config get dns","info":"8.8.8.8 8.8.4.4","code":0}"

config get date

Function

Obtain the current date/time of the CBOX.

Example command & feedback response:

config get date

{"cmd":"config get date","info":"2021-12-15 04:31:32","code":0}

config get timezone

Function

Retrieve the timezone configuration of the CBOX

Example command & feedback response:

config get timezone

{"cmd":"config get timezone","info":"UTC+0","code":0}

config get ntp

Function

Retrieve the current NTP settings of the CBOX.

Example command & feedback response:

config get ntp

{"cmd":"config get ntp","info":"0.north-america.pool.ntp.org/1.north-america.pool.ntp.org/2.north-america.pool.ntp.org/3.north-america.pool.ntp.org","code":0}

config set ip4addr {autoip}{dhcp}{static:[IPAddress]:[SubnetMask]:[DefaultGateway]}

Function

Used to configure the IP settings of the AV NETWORK port of the CBOX (the port used to control/monitor MXNet devices).

Example command & feedback response:

config set ip4addr autoip

"{"code":0,"cmd":"config set ip4addr autoip"}"

config set ip4addr dhcp

"{"code":0,"cmd":"config set ip4addr dhcp"}"

config set ip4addr static:192.168.1.120:255.255.255.0:192.168.1.1

{"code":0,"cmd":"config set ip4addr static:192.168.1.120:255.255.255.0:192.168.1.1"}

config set ip4addr2 {autoip}{dhcp}{static:[IPAddress]:[SubnetMask]:[DefaultGateway]}

Function

Configure the IP settings of the MENTOR/LAN port of the CBOX (the port used for accessing MENTOR/communicating to the CBOX from a 3rd party control system).

Example command & feedback response:

config set ip4addr autoip

"{"code":0,"cmd":"config set ip4addr autoip"}"

config set ip4addr dhcp

"{"code":0,"cmd":"config set ip4addr dhcp"}"

config set ip4addr static:192.168.1.120:255.255.255.0:192.168.1.1

{"code":0,"cmd":"config set ip4addr static:192.168.1.120:255.255.255.0:192.168.1.1"}

config set webloginpasswd {username:password}

Function

Set the admin username and password that is used to log in to the MENTOR web GUI of the CBOX.

Example command & feedback response:

config set webloginpasswd admin:admin

"{"code":0,"cmd":"config set webloginpasswd admin:admin"}"

config set restorefactory

Function

Upon execution, will restore the CBOX back to its factory default settings and clears all settings stored.

Example command & feedback response:

config set restorefactory

"OK"

config set reboot

Function

Reboots the CBOX upon command execution.

Example command & feedback response:

config set reboot

"{"code":0,"cmd":"config set reboot"}"

config set rs-232 alias on/off

Function

Enables or disables the RS-232 port of the CBOX.

Example

config set rs-232 alias on

"{"code":0,"cmd":"config set rs-232 alias on"}"

config set dns {nameserverip1} [nameserverip2]

Function

Set the DNS server configuration for the CBOX.

Example

config set dns 8.8.8.8 8.8.4.4

"{"code":0,"cmd":"config set dns 8.8.8.8 8.8.4.4"}"

OR

config set dns 8.8.8.8

“{"code":0,"cmd":"config set dns 8.8.8.8"}”

config set microusbmode {modeType}

NOTE: Depreciated.

Function

Note: Only for legacy CBOXes. Switches microUSB port’s communication between CBOX Linux shell and the built-in switch.

modeType = [1,2], 1:CBOX Linux Shell, 2:Switch.

Example

config set microusbmode 1

"{"cmd":"config set microusbmode 1","info":"OK","code":0}"

config set date {year} {month} {day} {hour} {minute} {second}

Function

Set the current date and time for the CBOX to use when reporting feedback on the System Log page.

Example command and feedback response:

config set date 2021 12 15 8 12 9

"{"code":0,"cmd":"config set date 2021 12 15 8 12 9"}"

config set timezone {UTC+B/UTC-B}

Function

Set the timezone (UTC formats) of the CBOX.

The range of **B** is 0 to 12.

Example command and feedback response:

config set timezone UTC-3

"{"code":0,"cmd":"config set timezone UTC-3"}"

Or

config set timezone UTC+2

"{"code":0,"cmd":"config set timezone UTC+2"}"

config set ntp {ntpserverip1} [ntpserverip2] [ntpserverip3] [ntpserverip4] [ntpserverip5]

Function

Modify the ntp servers for the CBOX to use.
ntpserverip1 ntpserverip2 ntpserverip3 ntpserverip4 ntpserverip5 are the ntp server addresses. Up to 5 addresses can be set, separated by spaces.

Example command and feedback response:

config set ntp 0.north-america.pool.ntp.org 1.north-america.pool.ntp.org 2.north-america.pool.ntp.org 3.north-america.pool.ntp.org

"{"code":0,"cmd":"config set ntp 0.north-america.pool.ntp.org 1.north-america.pool.ntp.org 2.north-america.pool.ntp.org 3.north-america.pool.ntp.org"}"

config set previews [on,off]

Function

NOTE: Requires CBOX FW V2.34 or higher.

Enables or disables the previews and image caching for the AUTO-MATRIX tab on the CBOX MENTOR interface.

Note: while the command uses the arguments [on,off] to set the preview parameter, the argument is stored as “1” (on) or “0” (off).

Example command and feedback response:

config set previews off

{"code":0,"cmd":"config set previews off"}

config set previews on

{"code":0,"cmd":"config set previews on"}

config get previews

Function

Retrieve the current status of the preview image setting for the CBOX's AUTO-MATRIX tab in the MENTOR web interface.

States are stored as "0" (off) or "1" (on).

Example command and feedback response:

Off:

config get previews

{"cmd":"config get previews","info":0,"code":0}

On:

config get previews

{"cmd":"config get previews","info":1,"code":0}

CBOX Device Commands

Used for configuring MXNet encoders or decoders.

NOTE: most of the API commands below, for the **{device_id/device_mac}** argument, will accept the arguments "**ALL**" for ALL MXNet devices, "**ALLTX**" for all MXNet encoders, and "**ALLRX**" for all MXNet decoders.

config get device status {device_id/device_mac}

Function

Retrieves the current status operation of an MXNet device. To retrieve the status information of all MXNet devices, use the command argument "config get device status ALL".

The "status' information pull will tie to the current video input/output status, such as resolution timing & frequency, HDR metadata, audio format, HDCP input/output, etc.

For more detailed information about a device, please see the command "config get device info" below.

Example command and feedback response:

config get device status 188A6A0102A8

{"cmd":"config get device status 188A6A0102A8","info":{"188A6A0102A8":{"chroma":"YUV422","profile":0,"hpd":"HPD1","hdr":"HDR1","light":0,"colordepth":"8 Bit","speed":"1G","id":"TCL-V1","video":"3840X2160p/60Hz","audio":"PCM","connectedname":"65Q825","hdcp":"HDCP2","switchip":"","switchport":""}},"code":0}

config get device info {device_id/device_mac}

Function

Retrieve detailed information on the current configuration settings on a specific device, such as its current IP configuration, firmware version, current channel subscriptions (channels are assigned to an encoder, and a decoder will subscribe to that channel for each individual video,audio,USB etc stream), etc.

Example command and feedback response:

Encoder info pull:

config get device info 188A6A0067A2

{"cmd":"config get device info 188A6A0067A2","info":{"chipset":"1520","version":"3.39","edid":"2","mac":"188A6A0067A2","dtype":"ast152x","usbmode":0,"subnet":"255.255.0.0","hdcp":"0","ch":"0007","exaudiovolume":"100","cec":1,"rs232baudrate":"9600.8.0","rs232mode":"2","id":"EV1-Apple","ipmode":"autoip","is_host":1,"light":0,"online":1677873977,"state":"s_srv_on","gateway":"169.254.0.254","ip":"169.254.6.37","rs232responsetype":1},"code":0}

Decoder info pull:

config get device info 188A6A0102A8

{"cmd":"config get device info 188A6A0102A8","info":{"version":"4.22","mac":"188A6A0102A8","stretch":"2","dtype":"ast152x","ch_u":"0007","ch_v":"0007","ch_s":"0007","cec":1,"rs232mode":"2","ch_r":"0007","light":0,"subnet":"255.255.0.0","rotate":"0","gateway":"169.254.0.254","stream":"on","chipset":"1520","blackout":"off","hdcp":"0","pattern":"0","ch":"0000","usbmode":0,"ch_a":"0000","ipmode":"autoip","id":"TCL-V1","hdrmode":0,"ch_p":"0000","state":"s_srv_on","ch_c":"0007","online":1677874038,"ip":"169.254.4.178","rs232baudrate":"9600.8.0","video":{"frames_per_second":"0","height":"0","width":"0"},"rs232responsetype":1},"code":0}

config set device reboot {device_id/device_mac}

Function

Reboot an MXNet encoder/decoder.

Example command and feedback response:

config set device reboot 188A6ACE87DC

"{"cmd":"config set device reboot 188A6ACE87DC","info":"","code":0}"

config set device restorefactory {device_id/device_mac}

Function

Restores the MXNet device specified in the {device_id/device_mac} argument to factory defaults.

Example command and feedback response:

config set device restorefactory 188A6ACE87DC

"{"cmd":"config set device restorefactory 188A6ACE87DC","info":"Reset to factory default...\nCurrent CRC is incorrect!!! Clear

All.\ndone\n","code":0}"

config set device id {new_id} {device_id/device_mac}

Function

Set the MXNet device's ID (also called the Custom Name of an MXNet Device) to a new ID specified in the {new_id} command argument. **Note: The new id for an encoder/decoder CANNOT contain a colon ':' or a ',' . It also CANNOT be named ALL/ALLRX/ALLTX.**

Example

config set device id TX66 188A6ACE87DC

"{"code":0,"cmd":"config set device id TX66 188A6ACE87DC"}"

config set device ip {autoip/dhcp/static:192.168.100.1:255.255.255.0:192.168.100.254} {device_id/device_mac}

Function

Changes the current IP configuration of an MXNet Encoder/Decoder. Default is "autoip".

WARNING, DRAGONS AHEAD!!

Proceed with caution when using this command. If an encoder/decoder is set to DHCP without a DHCP server connected to the switch, the CBOX **will be unable to re-discover it, and the encoder/decoder will need to be accessed via a physical USB connection in order to reset the settings**. An encoder/decoder set to a static IP that's different than the CBOX's "AV NETWORK" port's IP scheme **will also be lost until the "AV NETWORK" port's IP Scheme is changed to match it.**

Example command and feedback response:

config set device ip autoip 188A6ACE87DC

"{"cmd":"config set device ip autoip 188A6ACE87DC","info":"","code":0}"

config set device rm {device_id/device_mac}

Function

Removes an encoder/decoder from the CBOX's database (device list).

Example command and feedback response:

config set device rm 188A6ACE87DC

"{"code":0,"cmd":"config set device rm 188A6ACE87DC"}"

config set device audio input type hdmi/analog/auto {TxID/TxMAC}

Function

Changes an encoder's audio input selection. Applies to an encoder ONLY. Note: After switching the audio input selection, the device will automatically restart.

auto: The default setting. Defaults to use HDMI audio for the encoder's audio stream, but when a connection is detected on an encoder's ANALOG IN port, analog audio will be automatically selected to output over the encoder's audio stream.

hdmi: Will broadcast the audio stream using the incoming HDMI input's audio feed ONLY. Any ANALOG IN connection made will be ignored.

analog: Will broadcast the audio stream using the incoming ANALOG IN port's audio feed ONLY. Any HDMI input stream will be ignored, even when no ANALOG IN connection is established.

Example command and feedback response:

config set device audio input type hdmi 188A6ACE87DC

"{"cmd":"config set device audio input type hdmi 188A6ACE87DC","info":"Device will reboot","code":0}"

config set device audio volume {value} {decoder_id/decoder_mac}

Function

Adjust the **analog** audio output on an MXNet **Decoder**.

Value can be -1,0,1,2 ~,100.

Range from 0 to 100 %.

Default value '-1' means that the volume is handled by the FPGA, tries to mimic the volume level of the incoming audio stream.

Example command and feedback response:

config set device audio volume 6 188A6ACE87DC

"{"cmd":"config set device audio volume 6 188A6ACE87DC","info":"","code":0}"

config set device multicast on/off {device_id/device_mac}

Function

Changes the device from operating in multicast mode (**default/on)** to unicast mode (**off)** and vice versa.

Example command and feedback response:

config set device multicast on 188A6ACE87DC

"{"cmd":"config set device multicast on 188A6ACE87DC","info":"","code":0}"

config set device channel {ch_select} {TxID/TxMAC}

Function

Configure an encoder's channel to be a specified number in the {**ch_select}** argument. Note: **{ch_select}**'s scope is 0000 to 9999.

Example command and feedback response:

config set device channel 1006 188A6ACE87DC

"{"code":0,"cmd":"config set device channel 1006 188A6ACE87DC"}"

config set device description {description_may_include_space} {device_id/device_mac}

Function

Modify the description of an MXNet Endpoint to appear on the web GUI.

Example command and feedback response:

config set device description LEFT SCREEN OF CORNER 188A6ACE87DC

"{"code":0,"cmd":"config set device description LEFT SCREEN OF CORNER 188A6ACE87DC"}"

config set device video {width height fps} {RxID/RxMAC}

Function

Adjust the scaling output of an MXNet Decoder. To set the decoder back to "PASS-THROUGH" video output mode, enter in the command: config set device video 0 0 0 {RxID/RxMAC}.

Supported scaled resolution outputs for MXNet Decoders:

pass-through: 0 0 0

1280X720 50

1280X720 60

1920X1080 24

1920X1080 50

1920X1080 60

3840X2160 30

3840X2160 60

Example command and feedback response:

config set device video 1920 1080 60 188A6A45C4A5

"{"code":0,"cmd":"config set device video 1920 1080 60 188A6A45C4A5"}"

config set device edid {edidIndex} {TxID/TxMAC}

Function

Set encoder device's EDID.

Note, an MXNet Encoder must be on RS232 Mode 2 in order to control the EDID selection.

For AC-MXNET-1G-E, AC-MXNET-1G-AVDM-E, AC-MXNET-1G-EWP, AC-MXNET-1G-DANTE-E devices:

{edidIndex} = [1-15]

1: 1080P_6CH,

2: 1080P_3D_2CH,

3: 1080P_3D_6CH,

4: 4K30Hz_3D_2CH.

5: 4K30Hz_3D_6CH,

6: 4K30Hz_3D_8CH,

7: 1080P_2CH_HDR,

8: 1080P_6CH_HDR,

9: 1080P_3D_2CH_HDR,

10: 1080P_3D_6CH_HDR,

11: 4K30Hz_3D_2CH_HDR.

12: 4K30Hz_3D_6CH_HDR,

13: 4K30Hz_3D_8CH_HDR,

14: 1920X1200_2D_2CH_HDR.

15: User_EDID

For AC-MXNET-1G-EV2, AC-MXNET-1G-EV2WP, AC-MXNET-1G-AVDM-EV2 devices:

{edidIndex} = [1-15]

1: 1080P_6CH,

2: 1080P_3D_2CH,

3: 1080P_3D_6CH,

4: 4K30Hz_3D_2CH.

5: 4K30Hz_3D_6CH,

6: 4K30Hz_3D_8CH,

7: 1080P_2CH_HDR,

8: 1080P_6CH_HDR,

9: 1080P_3D_2CH_HDR,

10: 1080P_3D_6CH_HDR,

11: 4K60Hz_3D_2CH_HDR.

12: 4K60Hz_3D_6CH_HDR,

13: 4K60Hz_3D_8CH_HDR,

14: 1920X1200_2D_2CH_HDR.

15: User_EDID

Example command and feedback response:

config set device edid 3 188A6ACE87DC

"{"cmd":"config set device edid 3 188A6ACE87DC","info":"Set EDID success","code":0}"

config set device cec {hexData}[,{hexData},{hexData}...] {device_id/device_mac}

Function

Send CEC data to an MXNet endpoint. If the argument {hexData} = “poweron” or “poweroff”, The CBOX will send a large amount of most commonly used CEC power off/power on commands to the sink device from the specified Endpoint.

{hexData} argument is all in one “block”, no spaces, example: 0036

The command allows support to send multiple CEC commands at once to a sink device from a specified endpoint, **separated by ‘,’.**

Example: config set device cec 0036, 0037 188A6A45C4A5

Supports sending CEC commads to multiple MXNet endpoints (less than 50, **separated by colon :**) at once, using MAC address or CUSTOM NAME, or “ALLRX”, “ALLTX”, “ALL” arguments.

Example

config set device cec poweron 188A6A45C4A5

“{"cmd":"config set device cec poweron 188A6A45C4A5","info":"OK","code":0}”

or:

config set device cec poweroff 188A6A45C4A5

“{"cmd":"config set device cec poweroff 188A6A45C4A5","info":"OK","code":0}”

or:

config set device cec 0036 188A6A45C4A5

"{"cmd":"config set device cec 0036 188A6A45C4A5","info":"OK","code":0}"

or:

config set device cec poweron 188A6A45C4A5:188A6A45C4A7

“{"cmd":"config set device cec poweron 188A6A45C4A5:188A6A45C4A7","info":"OK","code":0}”

or:

config set device cec poweron ALLRX

“{"cmd":"config set device cec poweron ALLRX","info":"OK","code":0}”

config set device rs232mode {modeType} {device_id/device_mac}

Function

Configure the RS232 mode of an MXNet endpoint.

After switching the mode, the specified MXNet endpoint will automatically restart.

Note: When the RS232 mode is set to mode 1, the CBOX loses the ability to talk to the MCU of the MXNet endpoint, losing features such as EDID management, HDR metadata selection, light control, firmware updates, etc.

{modeType} = [1,2].

1: rs232 transparent transmission mode, data is transmitted from TX(RX) to RX(TX). Allows direct RS-232 passthrough from an MXNet Encoder to MXNet Decoder and vice versa.

2: rs232 guest mode, data is transmitted from the CBOX to an Encoder or Decoder (default mode).

Example

config set device rs232mode 2 188A6A45C4A5

"{"cmd":"config set device rs232mode 2 188A6A45C4A5","info":"Config success, 188A6A45C4A5 will reboot","code":0}"

config set device rs232setting {xx yy zz aa bb} {device_id/device_mac}

Function

Configure an MXNet endpoint's RS-232 port settings. Supports the ability to send rs232 setting data to multiple endpoints (less than 50, separated by colon), using MAC address or CUSTOM NAME, or "ALLRX", "ALLTX", or "ALL" arguments.

Note: For RS-232 communication/configuration from the CBOX, the specified endpoint **must** be in RS-232 mode 2.

xx = baudrate settings. Accepted values are a range from **300 ~ 115200**. Default is **9600**.

yy = data bits settings. Accepted values are **7** or **8**. Default is **8**.

zz = parity settings. Accepted values are: "**0"** (None), "**1"** (Even), "**2"** (Odd). Default is **none**.

aa = stop bit setting. Accepted value is only "**1"**

bb = flow control setting. Accepted value is only **"0"** (none).

Example command and feedback response:

config set device rs232setting 9600 8 0 1 0 188A6A45C4A5

"{"cmd":"config set device rs232setting 9600 8 0 1 0 188A6A45C4A5","info":"OK","code":0}"

or:

config set device rs232setting 9600 8 0 1 0 188A6A45C4A5:188A6A45C4A6

"{"cmd":"config set device rs232setting 9600 8 0 1 0 188A6A45C4A5:188A6A45C4A6","info":"OK","code":0}"

or:

config set device rs232setting 9600 8 0 1 0 ALLTX

"{"cmd":"config set device rs232setting 9600 8 0 1 0 ALLTX","info":"OK","code":0}"

config set device rs232responsetype [dataType] {TxID/TxMAC/TxID1:TxID2:TxID3:.../TxMAC1:TxMAC2:TxMAC3:.../RxMAC/ALLTX/ALLRX/ALL}

Function

Adjust the RS-232 feedback encapsulation for the specified MXNet endpoint(s).

The accepted values for the [dataType] argument are **“1”** (Base64), **“2”** (ASCII), **“3”** (HEX). The default value is **“1”** (Base64).

Supports the ability to send to multiple endpoints (less than 50, separated by colon :), using MAC address or CUSTOM NAME, or “ALLRX”, “ALLTX”, or “ALL” arguments.

Note: The RS-232 feedback will always be encoded in the specified format, whether the RS-232 data was sent in ASCII or HEX.

Example command and feedback response:

Note: The below feedback examples were using a Decoder’s RS-232 port connected via wiring the TX pin to RX, looping back any command being sent.

config set device rs232responsetype 1 188A6A0102A8

{"cmd":"config set device rs232responsetype 1 188A6A0102A8","info":"","code":0}

Example feedback response when executing the command “config set device rs232 1 TEST 188A6A0102A8”:

{"cmd":"config set device rs232 1 TEST 188A6A0102A8","info":"OK","code":0}

{"info":"VEVTVA==","id":"TCL-V1","source":"rs232","cmd":"","code":0,"mac":"188A6A0102A8"}

config set device rs232responsetype 2 188A6A0102A8

{"cmd":"config set device rs232responsetype 2 188A6A0102A8","info":"","code":0}

Example feedback response when executing the command “config set device rs232 2 54 45 53 54 188A6A0102A8”

{"cmd":"config set device rs232 2 54 45 53 54 188A6A0102A8","info":"OK","code":0}

{"info":"TEST","id":"TCL-V1","source":"rs232","cmd":"","code":0,"mac":"188A6A0102A8"}

config set device rs232responsetype 3 188A6A0102A8

{"cmd":"config set device rs232responsetype 3 188A6A0102A8","info":"","code":0}

Example feedback response when executing the command “config set device rs232 1 TEST 188A6A0102A8”

{"cmd":"config set device rs232 1 TEST 188A6A0102A8","info":"OK","code":0}

{"info":"54455354","id":"TCL-V1","source":"rs232","cmd":"","code":0,"mac":"188A6A0102A8"}

config set device rs232 dataType {serial port data} {device_id/device_mac}

Function

Tells the CBOX to encapsulate RS-232 data and outputs said data out of the specified MXNet endpoint(s) RS-232 port.

Note: For RS-232 communication/configuration from the CBOX, the specified endpoint must be in RS-232 mode 2.

The accepted values for argument "**datatype**" are **"1" or "2".** This specifies what type of RS-232 data is desired for transmission output. 1 equals ASCII data, 2 equals HEX.

The argument **{serial port data}** is the actual desired RS-232 data to be transmitted out of the specified MXNet Endpoint's RS-232 port. The value for this argument can contain spaces, and can be terminated with a carriage return **\r**, or newline feed **\n.**

RS-232 data can be transmitted to multiple MXNet endpoints (less than 50, **separated by colon :**) at the same time, using MAC address or CUSTOM NAME, or "ALLRX", "ALLTX", "ALL" arguments.

For feedback examples, please see the "config set device rs232responsetype" API command description above.

Example command and feedback response:

config set device rs232 1 SerialDATA \r 188A6A45C4A5

"{"cmd":"config set device rs232 1 SerialDATA \\r 188A6A45C4A5","info":"OK","code":0}"

config set device rs232 2 2F 36 F8 188A6A45C4A5

"{"cmd":"config set device rs232 2 2F 36 F8 188A6A45C4A5","info":"OK","code":0}"

config set device rs232 2 2F 36 F8 188A6A45C4A5:188A6A45C4A6

"{"cmd":"config set device rs232 2 2F 36 F8 188A6A45C4A5:188A6A45C4A6","info":"OK","code":0}"

config set device rs232 2 2F 36 F8 ALLRX

"{"cmd":"config set device rs232 2 2F 36 F8 ALLRX","info":"OK","code":0}"

config set device light on {device_id/device_mac}

Function

Enables the OLED, Status monitor, and RJ45 indicator lights on a specified Endpoint. NOTE: The endpoint's RS-232 mode must be set to Type2 in order for this to function.

Value for "ON" is stored as "0" for the "light" variable, can be queried via "config get device info" command.

Example command and feedback response:

config set device light on 188A6A45C4A5

"{"cmd":"config set device light on 188A6A45C4A5","info":"OK","code":0}"

config set device light off {device_id/device_mac}

Function

Set device's light off. NOTE: The endpoint's RS-232 mode must be set to Type2 in order for this to function.

Value for "OFF" is stored as "2" for the "light" variable, can be queried via "config get device info" command.

Example command and feedback response:

config set device light off 188A6A45C4A5

"{"cmd":"config set device light off 188A6A45C4A5","info":"OK","code":0}"

config set device light flash {device_id/device_mac}

Function

Set device's light flash. NOTE: The endpoint's RS-232 mode must be set to Type2 in order for this to function.

Value for "FLASH" is stored as "1" for the "light" variable, can be queried via "config get device info" command.

Example command and feedback response:

config set device light flash 188A6A45C4A5

"{"cmd":"config set device light flash 188A6A45C4A5","info":"OK","code":0}"

config set device capture {device_id/device_mac}

Function

Pulls an image capture from an encoder to be previewed.

NOTE: Depreciated, no longer needed. Previews can be accessed via navigation to this URL:

http://CBOX's_IP_address:81/capture.bmp?dev=encoder_device_mac

config set device osd on/off {RxID/TxMAC}

Function

Enables or disables the On-Screen-Display of an MXNet decoder. The OSD can be adjusted for multiple decoders(less than 50, **separated by colon**), using MAC address or CUSTOM NAME, or by using the argument “ALLRX”.

Stored as “0” (OFF) or “1” (ON) in the “osd” variable, queried from the “config get device info” command.

Example command and feedback response:

config set device osd on 188A6A45C4A5

"{"cmd":"config set device osd on 188A6A45C4A5","info":"OK","code":0}"

or:

config set device osd on 188A6A45C4A5:188A6A45C4A7

"{"cmd":"config set device osd on 188A6A45C4A5:188A6A45C4A7","info":"OK","code":0}"

or:

config set device osd on ALL

"{"cmd":"config set device osd on ALL","info":"OK","code":0}"

or:

config set device osd off 188A6A45C4A5

"{"cmd":"config set device osd off 188A6A45C4A5","info":"OK","code":0}"

config get device status {device_id/device_mac}

Function

Obtain diagnostic information/current routing status of an MXNet endpoint. To query all device’s of their status information, please use the command arguments “ALL”, “ALLRX”, or “ALLTX”.

Example command and feedback response:

config get device status 188A6A0102A8

{"cmd":"config get device status 188A6A0102A8","info":{"188A6A0102A8":{"chroma":"RGB","profile":0,"hpd":"HPD1","hdr":"HDR0","light":0,"colordepth":"8Bit","speed":"1G","id":"TCL-V1","video":"1920X1080p/59Hz","audio":"PCM","connectedname":"65Q825","hdcp":"HDCP ON","switchip":"","switchport":""}},"code":0}

config set device hdrmode [0,1] {RxID/RxMAC}

Function

Configure an MXNet decoder to either retain or remove HDR metadata when a decoder is set to a scaled output resolution.

0 = REMOVE

1 = RETAIN

Example command and feedback response:

config set device hdrmode 1 188A6A45C4A5

"{"code":0,"cmd":"config set device hdrmode 1 188A6A45C4A5"}"

config set device copyedid {RxID} {TxID }

Function

Copies the EDID of a connected sink device on a specified Decoder to store it in the USER EDID buffer of a specified Encoder.

Example command and feedback response:

config set device copyedid 188A6A45C4A5 188A6A0F4485

"{"cmd":"config set device copyedid 188A6A45C4A5 188A6A0F4485","info":"Copy success","code":0}"

config set device copyloopedid {TxID/TxMAC}

Function

Copies the EDID from a connected sink devices on an Encoder's HDMI LOOPOUT port, and stores the EDID to the USER EDID buffer of a specified encoder. NOTE: The Encoder's RS-232 mode needs to be set to Type2 in order for this command to function.

Example command and feedback response:

config set device copyloopedid 188A6A0F4485

"{"cmd":"config set device copyloopedid 188A6A0F4485","info":"Copy success","code":0}"

config set device profile [0,1] {TxID/TxMAC}

Function

Adjust the bandwidth usage of an encoder. Useful for encoder's connected to lower-grade CAT cables.

0 = auto, will try to use the maximum amount of bandwidth available (capped at 1G) **(default)**

1 = throttle the encoder to only use 200mb of total bandwidth.

Note: The specified encoder will reboot when this command is executed.

Example command and feedback response:

config set device profile 0 188A6A0F4485

"{"cmd":"config set device profile 0 188A6A0F4485","info":"188A6A0F4485 will reboot","code":0}"

config set device stretch [1,2] {RxID/RxMAC}

Function

Adjust a decoder's video output to either maintain the incoming signal aspect ratio/resolution, or stretch the incoming video signal to fit the screen of a connected sink device.

1 = stretch the signal

2 = maintain signal aspect ratio **(default)**

Example command and feedback response:

config set device stretch 2 188A6A45C4A5

"{"code":0,"cmd":"config set device stretch 2 188A6A45C4A5"}"

config set device rotate XX {RxID/RxMAC}

Function

Configure a decoder to rotate the image output by a specified degree.

Argument XX can equal "0" (no rotation, **default**), "3" or "180" (180 degree rotation),"5" or "90"(90 degree rotation) and "6" or "270" (270 degree rotation)

[XX=0,3,6 or 0,180,270], 0=rotate 0, 3=rotate 180, 5 = rotate 90. 6=rotate 270.

Example command and feedback response:

config set device rotate 3 188A6A45C4A5

"{"code":0,"cmd":"config set device rotate 3 188A6A45C4A5"}"

config set device hdcp XX {RxID/RxMAC}

Function

Modify a decoder's HDCP output.

Argument "XX" can equal "1" (passthrough, follow source), "3" (force HDCP 1.4, **default**), or "4" (force HDCP 2.2)

Example command and feedback response:

config set device hdcp 1 188A6A45C4A5

"{"cmd":"config set device hdcp 1 188A6A45C4A5","info":"Set HDCP success","code":0}"

Config set device hdcprequest [0,1] {EncoderID/EncoderMAC}

Function

Modify the HDCP request parameters for an MXNet encoder device.

Accepted arguments are 0,1. 0 = AUTO (HDCP 2.2) (**DEFAULT**). 1 = Do Not Request HDCP.

Can be queried using “config get device info” command, stored under “hdcp” variable.

Example command and feedback response:

config set device hdcprequest 1 188A6A010A8F

{"cmd":"config set device hdcprequest 1 188A6A010A8F","info":"Set HDCP success","code":0}

config set device videopath {TxID/TxMAC} {RxID/RxMAC}

Function

Configure the video stream route subscription from the specified MXNet encoder to the specified MXNet decoder.

Example command and feedback response:

config set device videopath 188A6A0F4485 188A6A45C4A5

"{"code":0,"cmd":"config set device videopath 188A6A0F4485 188A6A45C4A5"}"

config set device videopathdisable {RxID/RxMAC}

Function

Clears/De-Routes the incoming video stream subscription for the specified MXNet decoder.

Example command and feedback response:

config set device videopathdisable 188A6A45C4A5

"{"code":0,"cmd":"config set device videopathdisable 188A6A45C4A5"}"

config set device audiopath {TxID/TxMAC} {RxID/RxMAC}

Function

Configure the audio stream route subscription from the specified MXNet encoder to the specified MXNet Decoder.

Example command and feedback response:

config set device audiopath 188A6A0F4485 188A6A45C4A5

"{"code":0,"cmd":"config set device audiopath 188A6A0F4485 188A6A45C4A5"}"

config set device audiopathdisable {RxID/RxMAC}

Function

Clears/De-Routes the current incoming audio stream for the specified MXNet Decoder.

Example command and feedback response:

config set device audiopathdisable 188A6A45C4A5

"{"code":0,"cmd":"config set device audiopathdisable 188A6A45C4A5"}"

config set device rs232path {TxID/TxMAC} {RxID/RxMAC}

Function

Configure the RS-232 route from the specified MXNet encoder to the specified MXNet decoder

NOTE: Can only be used when both desired MXNet encoder and decoder's RS-232 mode is set to Type1.

Example command and feedback response:

config set device rs232path 188A6A0F4485 188A6A45C4A5

"{"code":0,"cmd":"config set device rs232path 188A6A0F4485 188A6A45C4A5"}"

config set device rs232pathdisable {RxID/RxMAC}

Function

Clears/De-Routes the RS-232 passthrough configuration subscription for the specified decoder.

NOTE: Can only be used when the devices' RS-232 mode is set to Type1.

Example command and feedback response:

config set device rs232pathdisable 188A6A45C4A5

"{"code":0,"cmd":"config set device rs232pathdisable 188A6A45C4A5"}"

config set device irpath {TxID/TxMAC} {RxID/RxMAC}

Function

Configure the IR Passthrough stream output subscription from a specified MXNet encoder to the Specified MXNet decoder.

Example command and feedback response:

config set device irpath 188A6A0F4485 188A6A45C4A5

"{"code":0,"cmd":"config set device irpath 188A6A0F4485 188A6A45C4A5"}"

config set device irpathdisable {RxID/RxMAC}

Function

Clears the route for IR Passthrough on the specified MXNet decoder.

Example command and feedback response:

config set device irpathdisable 188A6A45C4A5

"{"code":0,"cmd":"config set device irpathdisable 188A6A45C4A5"}"

config set device usbpath {TxID/TxMAC} {RxID/RxMAC}

Function

Configure the USB stream route from the specified MXNet decoder to the specified MXNet encoder.

Example command and feedback response:

config set device usbpath 188A6A0F4485 188A6A45C4A5

"{"code":0,"cmd":"config set device usbpath 188A6A0F4485 188A6A45C4A5"}"

config set device usbpathdisable {RxID/RxMAC}

Function

Clears the USB Route subscription for the specified MXNet decoder.

Example command and feedback response:

config set device usbpathdisable 188A6A45C4A5

"{"code":0,"cmd":"config set device usbpathdisable 188A6A45C4A5"}"

config set device ir XX {TxID/TxMAC}/{RxID/RxMAC}

Function

Send encapsulated IR codes to the specified MXNet endpoint. Supports Pronto and Global Cache code formats.

Argument “XX” is the IR Data desired to be sent to the MXNet endpoint. Supports two formats, Pronto and Global Cache.

Example for Pronto:

config set device ir 0000 0068 0024 0000 0155 00b2 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0042 0015 0043 0015 0043 0015 0043 0015 0043 0015 0043 0015 0043 0015 0043 0016 0042 0016 0016 0016 0042 0016 0016 0015 0043 0016 0016 0015 0043 0016 0016 0016 0016 0015 0043 0016 0016 0016 0042 0015 0017 0015 0043 0016 0016 0015 0043 0016 062c 0156 0058 0016 07c5 188A6A45C4A5.

Example for Global Cache:

Config set device ir 40000,1,1,344,180,20,23BBBBBBB,20,67,20,68CC,19,68CCDBBCCCB,19,23BCDBFBCC D,20,1594,343,90,21,2006 188A6A45C4A5

IR Codes can be sent to multiple device(less than 50, **separated by colon**) simultaneously, using MAC address or CUSTOM NAME, or ALLRX, ALLTX, ALL arguments.

Example command and feedback response:

config set device ir 0000 0068 0024 0000 0155 00b2 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 188A6A45C4A5

"{"code":0,"cmd":"config set device ir 0000 0068 0024 0000 0155 00b2 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 188A6A45C4A5"}"

or:

config set device ir 0000 0068 0024 0000 0155 00b2 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 188A6A45C4A5:188A6A45C4A6

"{"code":0,"cmd":"config set device ir 0000 0068 0024 0000 0155 00b2 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 188A6A45C4A5:188A6A45C4A6"}"

or:

config set device ir 0000 0068 0024 0000 0155 00b2 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 ALLRX

"{"code":0,"cmd":"config set device ir 0000 0068 0024 0000 0155 00b2 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 0016 ALLRX"}"

config set device kvmroaming RX1ID,X1,Y1:RX2ID,X2,Y2[:RX3ID,X3,Y3...] RX7ID

Function

Set kvm roaming, the last RX device RX7ID is the Primary device, it need connect the keyboard and mouse device in the front panel of the RX, its postion is (0, 0). Other RX (RX1, RX2, RX3, RX4...) are the secondary devices.

Example command and feedback response:

config set device kvmroaming RX1ID,-1,0:RX2ID,1,0:RX3ID,0,1 RX7ID

""{"cmd":"config set device kvmroaming RX1ID,-1,0:RX2ID,1,0:RX3ID,0,1 RX7ID ","info":"Config success, RX7ID will reboot","code":0}""

config set device hpdrst {device_id or device_mac}

Function

Resets the HDMI 5V Hotplug on the specified MXNet endpoint (can be encoder or decoder).

Example command and feedback response:

config set device hpdrst 188A6A45C4A5

"{"code":0,"cmd":"config set device hpdrst 188A6A45C4A5"}"

config set device exmxmode x {TxID/TxMAC}

Function

Changes the current downmixing preset for the AC-MXNET-1G-AVDM-E (Balanced Audio Output Only).

x: (x=[1-7])

[1-STD FX - Default Mode], [2-Low Center +], [3-Mid Center +], [4-High Center +], [5-Middle FX - Recommended] **(Default)**, [6-Full FX], [7-Voice FX]

Note: AC-MXNET-1G-AVDM-E firmware version needs to be greater than or equal to 3.22 to support this function.

Supports the ability to send the command to multiple devices(less than 50, **separated by colon**), using MAC address or CUSTOM NAME, or ALLTX argument.

For more information in regards to downmixing presets, please see here.

Example command and feedback response:

config set device exmxmode 0 188A6A45C4A5

"{"code":0,"cmd":"config set device exmxmode 0 188A6A45C4A5"}"

config set device exmxmode 6 188A6A45C4A5

"{"code":0,"cmd":"config set device exmxmode 6 188A6A45C4A5"}"

config set device stream on/off {RxID/RxMAC or RxMAC1:RxMAC2:...}

Function

Enables or disables the HDMI output of an MXNet decoder.

Supports the ability to send stream on/off to multiple devices(less than 50, separated by colon) simultaneously, using MAC address or CUSTOM NAME.

NOTE: Turning the stream off an MXNet Decoder kills the 5V active connection on the HDMI output, which will prevent CEC commands from being able to be sent. This command also does NOT clear any route subscriptions of an MXNet decoder.

Example command and feedback response:

config set device stream on 188A6A45C4A5

"{"code":0,"cmd":"config set device stream on 188A6A45C4A5"}"

config set device stream off 188A6A45C4A5

"{"code":0,"cmd":"config set device stream off 188A6A45C4A5"}"

config set device avdmid {new_id} {TxID/TxMAC}

Function

Set device daughter-card avdm id to new_id (alias name). Only applicable for AC-MXNET-1G-AVDM-E/AVDM-EV2 units. The ID of the AVDM encoder should never need to be changed.

Example command and feedback response:

config set device avdmid AVDMTX66ID 188A6ACE87DC

"{"code":0,"cmd":"config set device avdmid AVDMTX66ID 188A6ACE87DC"}"

config set device avdmdes {new_description} {TxID/TxMAC}

Function

Set device daughter-card description to new_description. **Only applicable for AC-MXNET-1G-AVDM-E and AC-MXNET-1G-AVDM-EV2 units.**

Example command and feedback response:

config set device avdmdes AVDMTX66DES 188A6ACE87DC

"{"code":0,"cmd":"config set device avdmdes AVDMTX66DES 188A6ACE87DC"}"

config set device osdsize [1,2] {RxID/RxMAC}

Function

Adjust the resolution output of the FORCED IMAGE on an MXNet Decoder.

Argument “1” equals 640x480 **(default)**, argument “2” is 1920x1080. After setting, the decoder will automatically restart.

Example command and feedback response:

config set device osdsize 1 188A6A45C4A5

"{"code":0,"cmd":"config set device osdsize 1 188A6A45C4A5"}"

config set device osdsize 2 188A6A45C4A5

"{"code":0,"cmd":"config set device osdsize 2 188A6A45C4A5"}"

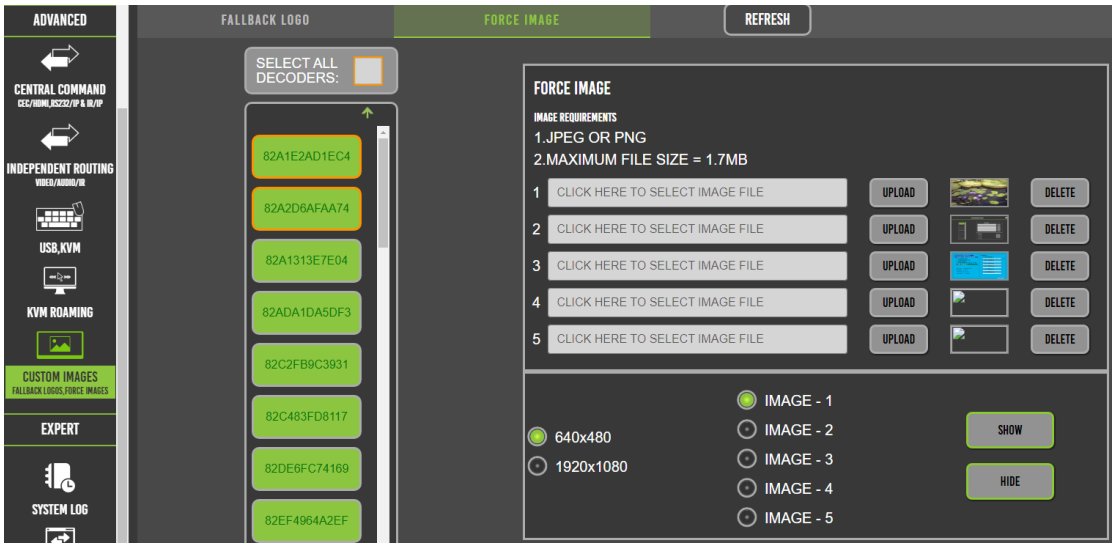
config set device displaypicture 1/2/3/4/5 {RxID/RxMAC}

Function

Forces the desired image onto a display from the desired MXNet Decoder.

Up to five pictures can be uploaded to the CBOX by the CBOX's Web GUI, under "CUSTOM IMAGES" tab.

The arguments "1","2","3","4", "5" specifies which image to be forced, corresponds with the Web GUI Screenshot below.



Example command and feedback response:

config set device displaypicture 1 188A6A45C4A5

"{"code":0,"cmd":"config set device displaypicture 1 188A6A45C4A5"}"

config set device displaypicture 2 188A6A45C4A5

"{"code":0,"cmd":"config set device displaypicture 2 188A6A45C4A5"}"

config set device blackout on/off {RxID/RxMAC}

Function

Creates a blackout image for the specified MXNet Decoder's output. The "Blackout" feature essentially captures a frame of the incoming video feed and paints it black. Audio will still be sent to the display, and the resolution of the blackout image will be whatever the incoming video timing's resolution is.

NOTE: Because it takes a still frame of an incoming video feed and colors it black, an incoming video stream is required to use this feature.

Example command and feedback response:

config set device blackout on 188A6A45C4A5

"{"code":0,"cmd":"config set device blackout on 188A6A45C4A5"}"

config set device blackout off 188A6A45C4A5

"{"code":0,"cmd":"config set device blackout off 188A6A45C4A5"}"

config set device quality [quality_mode] {TxID/TxMAC}

Function

Configures the video stream output of a specified MXNet Encoder.

The argument “quality_mode” can be the following: “0”, “-1”, “1”, “2”, “3”, “4”, “5”.

- ○ 0: Graphic mode. Picture quality first. May drop frame. **(default)**
- ○ -1: Video mode. Smooth frame rate first. Will change video quality dynamically.
- ○ Arguments “1” through “5” will fix the buffer to prioritize framerate, sacrificing quality the higher you go.

Example command and feedback response:

config set device quality -1 188A6A45C4A5

"{"code":0,"cmd":"config set device quality -1 188A6A45C4A5"}"

config set device quality 0 188A6A45C4A5

"{"code":0,"cmd":"config set device quality 0 188A6A45C4A5"}"

config set device osdmode 1/2/3 {RxID/RxMAC/RxID1:RxID2:RxID3:.../RxMAC1:RxMAC2:RxMAC3:.../ALLRX/ALL}

Function

Configure the On-Screen-Display mode of an MXNet Decoder.

Mode 1: Four lines, Decoder custom name, Decoder MAC address, Decoder IP address, Encoder custom name, on bottom-left of Display.

Mode 2: Two lines, Decoder custom name, Encoder custom name, on bottom-left of Display.

Mode 3: One line, Encoder custom name, on center-top of Display.

Support to set multiple device(less than 50, separated by colon), using MAC address or CUSTOM NAME, or ALL, ALLRX.

Example command and feedback response:

config set device osdmod 1 188A6ACE87DC

{"cmd":"config set device osdmode 1 188A6ACE87DC","info":"","code":0}

config set device osdfontsize [font size] {RxID/RxMAC/RxID1:RxID2:RxID3:.../RxMAC1:RxMAC2:RxMAC3:.../ALLRX/ALL}

Function

Adjust the font size of the OSD output.

Support to set multiple device(less than 50, separated by colon), using MAC address or CUSTOM NAME, or ALL, ALLRX.

NOTE: Font sizes greater than 40 may be too large and will bleed outside the display output.

Example command and feedback response:

config set device osdfontsize 32 188A6ACE87DC

{"cmd":"config set device osdfontsize 32 188A6ACE87DC","info":"","code":0}

config set device osdcolor [hex color] {RxID/RxMAC/RxID1:RxID2:RxID3:.../RxMAC1:RxMAC2:RxMAC3:.../ALLRX/ALL}

Function

Set decoder osd color. Argument [hex color] is the raw HEX color code desired.

Support to set multiple device(less than 50, separated by colon), using MAC address or CUSTOM NAME, or ALL, ALLRX.

Example command and feedback response:

config set device osdcolor 8DC63F 188A6ACE87DC

{"cmd":"config set device osdcolor 8DC63F 188A6ACE87DC","info":"","code":0}

config set device exaudio volume {VALUE} {TxID/TxMAC}

Function

Adjust the volume output of an MXNet Encoder's Extracted audio port.

Argument {VALUE} = [0~100].

Example

config set device exaudio volume 50 188A6ACE87DC

{"cmd":"config set device exaudio volume 50 188A6ACE87DC","info":"","code":0}

config set device pattern 0/1/2 {RxID/RxMAC}

Function

Enables/disables the built-in test pattern for an EV1 MXNet Decoder.

0: PATTERN OFF

1: COLORBAR 1080P60

2: COLORBAR 4K30.

Note: Not support on V2 device types. For V1 Decoders, when the decoder version is a V3.XX version, its version should be 3.32 and above. When the V1 Decoder firmware version is V4.XX, its version should be 4.18 and above.

Example command and feedback response:

config set device pattern 1 188A6ACE87DC

{"cmd":"config set device pattern 1 188A6ACE87DC","info":"","code":0}

config set device ediddata {EDID DATA} {TxID/TxMAC}

Function

Import the desired raw EDID data to an MXNet encoder, and stores said data in the encoder's USER EDID buffer.

{EDID DATA} is 256 bytes Hex EDID datas.

Example

config set device ediddata 00 ff ff ff ff ff ff 00 37 18 01 00 01 00 00 00 04 19 01 03 80 73 41 78 0a cf 74 a3 57 4c b0 23 09 48 4c 21 08 00 81 80 45 40 61 40 95 00 01 01 01 01 01 01 01 01 08 e8 00 30 f2 70 5a 80 b0 58 8a 00 50 1d 74 00 00 1e 02 3a 80 18 71 38 2d 40 58 2c 45 00 50 1d 74 00 00 1e 00 00 00 fc 00 4d 58 58 2d 4d 2d 33 0a 20 20 20 20 20 00 00 00 fd 00 18 4b 0f 87 3c 00 0a 20 20 20 20 20 20 01 df 02 03 3c f2 51 61 60 65 66 01 03 04 05 07 10 12 13 14 16 1f 20 22 23 09 07 07 83 01 00 00 e3 05 c0 00 6e 03 0c 00 11 00 08 3c 20 80 80 01 02 03 04 e2 0f 0f 67 d8 5d c4 01 78 80 03 8c 0a d0 8a 20 e0 2d 10 10 3e 96 00 c4 8e 21 00 00 18 8c 0a d0 90 20 40 31 20 0c 40 55 00 c4 8e 21 00 00 18 01 1d 00 bc 52 d0 1e 20 b8 28 55 40 c4 8e 21 00 00 1e 00 00 00 00 00 00 00 00 00 00 00 00 00 af 188A6ACE87DC

{"cmd":"config set device ediddata 00 ff ff ff ff ff ff 00 37 18 01 00 01 00 00 00 04 19 01 03 80 73 41 78 0a cf 74 a3 57 4c b0 23 09 48 4c 21 08 00 81 80 45 40 61 40 95 00 01 01 01 01 01 01 01 01 08 e8 00 30 f2 70 5a 80 b0 58 8a 00 50 1d 74 00 00 1e 02 3a 80 18 71 38 2d 40 58 2c 45 00 50 1d 74 00 00 1e 00 00 00 fc 00 4d 58 58 2d 4d 2d 33 0a 20 20 20 20 20 00 00 00 fd 00 18 4b 0f 87 3c 00 0a 20 20 20 20 20 20 01 df 02 03 3c f2 51 61 60 65 66 01 03 04 05 07 10 12 13 14 16 1f 20 22 23 09 07 07 83 01 00 00 e3 05 c0 00 6e 03 0c 00 11 00 08 3c 20 80 80 01 02 03 04 e2 0f 0f 67 d8 5d c4 01 78 80 03 8c 0a d0 8a 20 e0 2d 10

10 3e 96 00 c4 8e 21 00 00 18 8c 0a d0 90 20 40 31 20 0c 40 55 00 c4 8e 21 00 00 18 01 1d 00 bc 52 d0 1e 20 b8 28 55 40 c4 8e 21 00 00 1e 00 00 00 00 00 00 00 00 00 00 00 00 00 00 af 188A6ACE87DC","info":"","code":0}

Config set device useredid {EncoderID/MAC}

Function:

Reports the raw HEX data of the EDID currently stored in the desired encoder's USER EDID buffer.

config set device usbmode [0,1] {DeviceID/MAC}

Function

NOTE: Requires CBOX FW V2.34 or higher.

Configure the USB operation mode for Evolution Two (V2) encoder/decoder endpoints. NOTE: Only Applicable for V2 devices.

Argument "0" = Software-based USB mode. Does not allow for high-speed USB devices, such as camera systems, but allows KVM roaming.

Argument "1" (default) = Hardware-based USB mode. Allows for high-speed USB connectivity, such as a camera or microphone. Only allows for 1:1 USB connections from an encoder -> Decoder (one Decoder can subscribe to an encoder at a time). Only one high-speed USB connection can be established at a time for an encoder -> decoder pair.

NOTE: V2 Devices within the SN or MAC address range below can ONLY utilize hardware USB mode!!!!!!!

SN Range: 113322091600001~113322091601000

MAC Address Range: 188A6A00EC04~188A6A0OEFEB

Example command and feedback response:

config set device usbmode 1 188A6A0130A1

{"cmd":"config set device usbmode 1 188A6A0130A1","info":"EV2 will reboot","code":0}

When setting usbmode 0 on a device that doesn't support it:

config set device usbmode 0 188A6A00EFDF

{"error":"[string \"plugin.ast152x\"]:0: The mac of the device is between 188A6A00EC04 and 188A6A0OEFEB, only usbmode=1 is supported.","cmd":"config set device usbmode 0 188A6A00EFDF","code":-1}

config set device arp enable [EncoderID/MAC] [DecoderID/MAC]

Function:

NOTE: Requires CBOX V2.43 or higher. V2 Endpoints only!

Enables and routes the Audio Return Path (ARP) from a specified Decoder to the specified encoder. **Devices will reboot if ARP was initially disabled**

Will set device attributes "ARPon" to "y" (default is "n") and "ARPchannel" to the encoder's perspective channel.

Example command and feedback response:

config set device arp enable V2-INTEGRA V2-TCL

Feedback: {"cmd":"config set device arp enable V2-INTEGRA V2-TCL","info":"ARP successfully routed from V2-TCL to

V2-INTEGRA, channel 0001","code":0}

config set device arp disable [Encoder/DecoderID/MAC]

Function:

NOTE: Requires CBOX V2.43 or higher. V2 Endpoints only!

Disables Audio Return Path on the encoder/decoder. Changes “ARPon” attribute to “n”, “ARPchannel” attribute to “none” (default). **Device will reboot for changes to take effect.**

Example command and feedback response:

config set device arp disable V2-TCL

Feedback: {"cmd":"config set device arp disable V2-TCL","info":"ARP disabled. Device V2-TCL will now reboot.","code":0}

config set device arp input [hdmiout, spdif] [DecoderID/DecoderMAC]

Function:

NOTE: Requires CBOX V2.43 or higher. V2 Endpoints only!

Modifies the input port for a decoder’s Audio Return Path transmission.

"hdmiout" = changes Decoder’s ARP input to be the HDMI output port connected to an ARC/eARC display. (Default value stored in attribute "ARPinput")

"spdif" = changes Decoder's ARP input to the SPDIF input port of the decoder.

Command will change the attribute "ARPinput"'s value of the decoder to the respective chosen argument.

NOTE: The encoder will follow whatever settings the decoder is utilizing, nothing needs to be changed on the encoder.

Example command and feedback response:

“config set device arp input hdmiout V2-TCL”

Feedback: {"cmd":"config set device arp input hdmiout V2-TCL","info":"Device V2-TCL ARP input is now HDMI out.","code":0}

config set device srckeeptime [time] {DecoderID/DecoderMAC}

Function:

NOTE: Requires CBOX V2.55 or higher.

Adjusts the time the specified decoder will “hang on” to the last video image captured from an encoder before the reverts to the fallback “No Signal” image. Default time is 10 seconds. If the time is set to “0”, the decoder will immediately revert to the fallback no signal image when the video stream from an encoder is lost or removed.

Example command and feedback response:

config set device srckeeptime 0 188A6ACE87DC

Feedback: {"cmd":"config set device srckeeptime 0 188A6ACE87DC","info":"","code":0}

Config set device usb5v {0,1,2} [EncoderID/EncoderMAC] [DecoderID/DecoderMAC]

Function:

NOTE: Requires CBOX V2.55 or higher. V2 Decoders on PCB V0.6 only!

Adjusts the configuration for the USB 5v connection on V2 Decoders. NOTE: Only decoders on PCB V0.6 or higher supports this function, such as the AC-MXNET-1G-DV2-C!

Option “0” = Follow the encoder, targeted via the [EncoderID/EncoderMAC] argument. When the USB-B/USB HOST connection is dropped on the specified encoder, the specified decoder will shut off its USB 5v connection.

Option “1” = Always on (default). When option “1” is desired, the argument [EncoderID/EncoderMAC] is not required.

Option “2” = Always off. When Option “2” is desired, the argument [EncoderID/EncoderMAC] is not required.

Example command and feedback response:

config set device usb5v 0 188A6ACE87DB 188A6ACE87DC

{"cmd":"config set device usb5v 0 188A6ACE87DB 188A6ACE87DC","info":"","code":0}

config set device usb5v 1 188A6ACE87DC

{"cmd":"config set device usb5v 1 188A6ACE87DC","info":"","code":0}

Config set device kmoip [0,1] {DeviceID/DeviceMAC}

Function:

NOTE: Requires CBOX V2.55 or higher.

Enables or Disables the KM/IP feature on an MXNet endpoint. The KM/IP feature allows for software USB emulation (allowing a decoder to connect to multiple encoders for keyboard/mice support). **NOTE: Disabling this feature may improve functionality for certain USB devices, however it may also break functionality entirely for certain USB devices. Use this command at your own risk!**

Option “0” = Off.

Option “1” = On (default).

Example command and feedback response:

config set device kmoip 1 188A6ACE87DC

Feedback: {"cmd":"config set device kmoip 1 188A6ACE87DC","info":"","code":0}

config set device audiomute [0~10000] {DecoderID/DecoderMAC}

Function:

Adjusts the HDMI Output Audio Delay time. Delay time is measured in milliseconds. Default is 2000ms. NOTE: Certain devices have increased sensitivity when the audio format changes too quickly. Use this command to raise or lower the audio mute time if you’re experiencing a “popping” or “cracking” sound when the audio format changes.

Example command and feedback response:

config set device audiomute 2000 188A6ACE87DC

Feedback: {"cmd":"config set device audiomute 2000 188A6ACE87DC","info":"","code":0}

config get device audiomute [DecoderID/DecoderMAC]

Function:

Obtain the current audiomute value for the specified decoder.

Example command and feedback response:

config get device audiomute V1-BOTLEFT

Feedback: {"cmd":"config get device audiomute V1-BOTLEFT","info":2000,"code":0}

config set device sourceinput [1,2] {EncoderID/EncoderMAC}

Requires AC-MXNET-CBOX V2.71 / AC-MXNET-CBOX-B/HA V4.19 or higher.

Function:

This command is to adjust the active source input on encoders with multiple source inputs, such as switching between HDMI IN or USB-C IN. Currently only applicable on the AC-MXNET-1G-EV2WP device.

Example Command and Feedback Response:

config set device sourceinput 2 188A6A0249DB

Feedback: {"cmd":"config set device sourceinput 2 088A6A01021B","info":"","code":0}

config set device shareusb [0,1] {EncID/EncMAC/DecID/DecMAC/ALL/ALLTX/ALLRX}

Requires AC-MXNET-CBOX V2.71 / AC-MXNET-CBOX-B/HA V4.19 or higher.

Note: This command is only applicable on V2 endpoints running FPGA Firmware V9.18.9 or above!

Note: Any/All targeted device(s) will reboot after this command is executed.

Function:

Enables the USB Shared Mode on V2 endpoints. USB Shared Mode is a new feature that allows the connection of multiple decoders' USB streams (up to 7 decoders total) to a single encoder, and enables the user to connect multiple isochronous devices, such as USB cameras, from multiple decoders to one encoder. Note: This feature does NOT allow multiple USB Hubs connected to decoders to sync with the routed encoder. Only from a direct USB port connection on the decoder.

Option "**0**" == off. Default setting.

Option “**1**” == enable USB Shared Mode.

Example Command and Feedback Response:

config set device shareusb 1 ALL

Feedback: {"code":0,"cmd":"config set device shareusb 1 ALL"}

config set device ull {0,1} [DecoderID/DecoderMAC/ALLRX]

Requires AC-MXNET-CBOX V2.71 / AC-MXNET-CBOX-B/HA V4.19 or higher.

Note: This command is only applicable on V2 endpoints running FPGA Firmware V9.18.9 or above!

Note: Any/All targeted device(s) will reboot after this command is executed.

Function:

Enables Ultra Low-Latency Mode (ULL) on the targeted V2 Decoder(s). Ultra Low-Latency Mode is a new feature on V2 decoders that massively reduces the already low amount of latency that V2 endpoints have. Without ULL enabled, latency on V2 endpoints is equivalent to one frame (~19ms for 60hz, ~39ms for 30hz). With ULL enabled, latency is reduced to an average of **2-4ms** latency at 60hz, and **7-9ms** latency at 30hz.

WARNING: Enabling this on decoders that are generating videowalls may introduce a screen tearing/image blanking to the video stream due to the sensitivity of Genlock. Advise caution when enabling this feature combined with videowall generation.

CAUTION: This feature relies on a solid infrastructure with no traffic congestion. If the infrastructure isn’t solid or there is a heavy amount of traffic on the network, screen tearing/image blanking may occur.

Example Command and Feedback Response:

config set device ull 1 V2-DECODER

Feedback: {"cmd":"config set device ull 1 TCL","info":"","code":0}

config set device dspmode {1,2} [EncID/EncMAC/ALLTX]

Requires AC-MXNET-CBOX V2.71 / AC-MXNET-CBOX-B/HA V4.19 or higher.

Requires any encoder endpoint to be on the latest version of firmware on our KB here.

Function:

Changes the Digital Signal Processor’s operation mode. Used to resolve cases where extracted audio is lost over an extended period of time on specific sources.

1 == Default, normal operation.

2 == The DSP will wait to process extracted audio until 5 seconds after a hot plug event occurs, to ensure the audio signal is stabilized. This may introduce a slight delay to when the extracted audio is outputted when switching to this source.

Example command and feedback response:

Command: config set device dspmode {1,2} [EncID/EncMAC/ALLTX]

Response: {"cmd":"config set device dspmode 2 ALLTX"}

config get device routes [v,a,u,r,s] {DecID/DecMAC/ALLRX}

Requires AC-MXNET-CBOX V2.71 / AC-MXNET-CBOX-B/HA V4.19 or higher.

Function:

Used to obtain the current active routes on the specified decoder(s).

v = Active Video Stream

a = Active Audio Stream

u = Active USB Stream

r = Active IR stream

s = Active RS-232 stream

Various stream requests can be combined into a single command execution, for example, “config get device routes va OUTPUT1” will query the active video AND audio streams of the decoder “OUTPUT1” simultaneously. The “ALLRX” argument can be used to query the active routes of all decoders currently within the MXnet network.

Example Command and Feedback Response:

Command: config get device routes vaurs OUTPUT1

Response: {"info":{"OUTPUT1":{"ir":"none","usb":"INPUT2","audio":"INPUT1","video":"INPUT1","rs232":"none"}},"cmd":"config get device routes vaurs TCL"}

Command: config get device routes va ALLRX

Response:

{"info":{"OUTPUT1":{"video":"none","audio":"none"},"OUTPUT2":{"video":"INPUT1","audio":"INPUT1"},"OUTPUT3":{"video":"INPUT2","audio":"INPUT2"},"OUTPUT4":{"video":"INPUT1","audio":"INPUT2"},"OUTPUT5":{"video":"INPUT2","audio":"INPUT2"},"OUTPUT6":{"video":"INPUT3","audio":"none"},"OUTPUT6":{"video":"none","audio":"INPUT3"}},"cmd":"config get device routes va ALL"}

config set device audiosrc [1,2] {EncID/EncMAC/DecID/DecMAC}

Requires AC-MXNET-CBOX V2.71 / AC-MXNET-CBOX-B/HA V4.19 or higher.

This command may only be used by Dante V2 devices.

Function:

Used to change the audio output from MXnet V2 audio (argument option **1)** or Dante routed audio stream (argument option **2**).

Requires a restart to take effect.

CBOX Video Wall Commands

Note: These commands are depreciated, and it is recommended to use the ‘**vwid’** API command arguments.

Use for video wall setting.

vw list

Function

Get video wall list.

Example

vw list

{"cmd":"vw

list","info":{"vw1":{"cols":3,"rows":3,"osd":true,"hosts":["tx1"],"bezelgap":[1345,1350,645,650],"rotate":[],"clients":["rx1"]}},"code

":0}

vw get {vw_name}

Function

Get video wall {vw_name}.

Example

vw get vw1

{"cmd":"vw get

vw1","info":{"cols":3,"rows":3,"osd":true,"hosts":["tx1"],"bezelgap":[1345,1350,645,650],"rotate":[],"clients":["rx1"]},"code":0}

vw add {vw_name} {rows} {cols}

Function

Add video wall, total rows is {rows}, total columns is {cols}. **Note: vw_name can not contain a colon ':'.**

Example

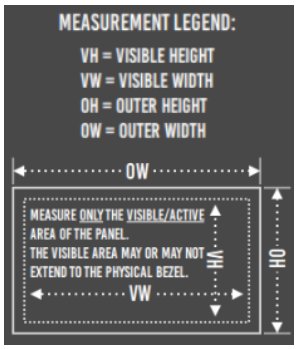
vw add vw1 3 3

{"code":0,"cmd":"vw add vw1 3 3"}

vw gap {vw_name} {vw} {ow} {vh} {oh}

Function

Set video wall bezel gap.



Example

vw gap vw1 1345 1350 645 650

{"code":0,"cmd":"vw gap vw1 1345 1350 645 650"}

vw tx {vw_name} {tx1} ..

Function

Set video wall tx1.

Example

vw tx vw1 tx1

{"code":0,"cmd":"vw tx vw1 tx1"}

vw rx {vw_name} {rx1[:row:col[:rotate]]} ..

Function

Set video wall {rx1} at row, col..

row>=1, col>=1, If the position row, col exist RX, new RX will replace old RX, and the old RX will be removed and displays an entire picture of TX. Rotate =0 or 180.

Example

vw rx vw1 rx1

{"code":0,"cmd":"vw rx vw1 rx1"}

vw osd {vw_name} {on/off}

Function

Enable or disable osd.

Example

vw osd vw1 on

{"code":0,"cmd":"vw osd vw1 on"}

vw rmtx {vw_name} {tx1} [tx2] ..

Function

Delete video wall tx1.

Example

vw rmtx vw1 tx1

{"code":0,"cmd":"vw rmtx vw1 tx1"}

vw rmrx {vw_name} {rx1} [rx2 rx3...]

Function

Delete video wall rx1, rx2, rx3....

Example

vw rmrx vw1 rx1

{"code":0,"cmd":"vw rmrx vw1 rx1"}

vw rm {vw_name}

Function

Delete video wall {vw_name}.

Example

vw rm vw1

{"cmd":"vw rm vw1","info":"OK","code":0}

vw active {vw_name} [force]

Function

Active video wall {vw_name}.

Example

vw active vw1

{"cmd":"vw active vw1","info":"OK","code":0}

IPC Matrix Commands

Use for matrix stream configurations.

matrix list

Function

Obtain a list of all available matrix configurations.

Example command and feedback response:

matrix list

{"cmd":"matrix list","info":{"mx1":{"type":"va","srcs":{"rx1":"tx1","rx2":"tx1"}}},"code":0}

matrix get {name}

Function

Obtain the information of a matrix list under the specified {name} argument.

Example command and feedback response:

matrix get mx1

{"cmd":"matrix get mx1","info":{"type":"va","srcs":{"rx1":"tx1","rx2":"tx1"}},"code":0}

matrix add {name} {video/audio/usb/infrared/serial/all}

Function

Add new matrix under the specified {name} argument. The associated parameters for all inbound streams are as follows: **video="v", usb="u", audio="a", infrared="r", serial="s", all="z", [""]="v"**. So, if we want to add a matrix named mx1, and use the streams Video (v) and Audio (v) for this matrix configuration, we can use parameters **av**.

Example command and feedback response:

matrix add mx1 va

{"code":0,"cmd":"matrix add mx1 va"}

matrix set {name} {tx1 rx1 rx2 .. rxn[, tx2 rx..]}

Function

Add Encoders and Decoders to the matrix configuration. Multiple transmitters can be added to the argument, examples below. Streams to be routed are set via “matrix add” command described above.

Example

matrix set mx1 tx1 rx1 rx2

{"cmd":"matrix set mx1 tx1 rx1 rx2","info":"OK","code":0}

matrix aset [[name]:[video/audio/usb/infrared/serial/all]] {tx1 rx1 rx2 .. rxn[, tx2 rx..]}

Function

parameters video is equal to v. (**video="v", usb="u", audio="a", infrared="r", serial="s", all="z", [""]="v"**). So, if we want to connect TX1's **video** and **audio** to RX1, we can use parameters **av**. matrix aset :av TX1 RX1 RX2 .. RXn. Reset tx1 point to rx1 and rx2, or reset the devices in the matrix[name]. The parameters in [] is **not necessary**. The default type is **video**. Effective immediately.

e.g.: set infrared from TX1 to RX1, RX3, ... RXn matrix aset :infrared TX1 RX1 RX2 .. RXn e.g.: set usb from TX1 to RX1, RX3, ... RXn matrix aset :usb TX1 RX1 RX2 .. RXn

Example

matrix aset tx1 rx1 rx2

{"cmd":"matrix aset tx1 rx1 rx2","info":"OK","code":0}

matrix rm {name}

Function

Delete matrix{name}.

Example

matrix rm mx1

{"cmd":"matrix rm mx1","info":"OK","code":0}

matrix active {name} [force]

Function

Active matrix {name}, if add [force], IPC will force refresh all the devices of matrix {name}.

Example

matrix active mx1

{"cmd":"matrix active mx1","info":"OK","code":0}

IPC Matrix Preset Commands

Used to add matrix presets that can be recalled at will. Presets made by this API command will also show in the web GUI, under the “AUTO-MATRIX” tab.
NOTE: Adding a matrix preset will save the current configuration of **all** routes on the MXNet system. It does not save specific encoder/decoder routes, and cannot be modified to do so.

Matrix preset list

Function

Obtain the list of currently created matrix presets.

Example command & feedback response (one preset “test” has been created for this example):

matrix preset list

{"cmd":"matrix preset list","info":{"test":{"OUTPUT1":{"vmode":"0","ch_v":"0001","dtype":"ast152x","ch_p":"0000","ch_a":"0001","ch_u":"0001","ch_s":"0004","ch_r":"0004"},"OUT3":{"vmode":"0","ch_v":"0001","dtype":"ast152x","ch_p":"0006","ch_a":"0001","ch_u":"0001","ch_s":"0004","ch_r":"0004"},"OUT2":{"vmode":"0","ch_v":"0001","dtype":"ast152x","ch_p":"0000","ch_a":"0001","ch_u":"0001","ch_s":"0004","ch_r":"0004"},"TCL-V2":{"vmode":"0","ch_v":"0005","dtype":"ast152x","ch_p":"0000","ch_a":"0005","ch_u":"0005","ch_s":"0000","ch_r":"0000"}}},"code":0}

matrix preset get {name}

Function

Recalls the information on a specific matrix preset that was already created in the system.

Example command & feedback response:

matrix preset get test

{"cmd":"matrix preset get test","info":{"MONTLEFT":{"vmode":"0","ch_v":"0001","dtype":"ast152x","ch_p":"0000","ch_a":"0001","ch_u":"0001","ch_s":"0004","ch_r":"0004"},"188A6A00869A":{"vmode":"0","ch_v":"0001","dtype":"ast152x","ch_p":"0006","ch_a":"0001","ch_u":"0001","ch_s":"0004","ch_r":"0004"},"MONTRIGHT":{"vmode":"0","ch_v":"0001","dtype":"ast152x","ch_p":"0000","ch_a":"0001","ch_u":"0001","ch_s":"0004","ch_r":"0004"},"TCL-V2":{"vmode":"0","ch_v":"0005","dtype":"ast152x","ch_p":"0000","ch_a":"0005","ch_u":"0005","ch_s":"0000","ch_r":"0000"}},"code":0}

matrix preset add {name}

Function

Adds a new matrix preset to the system with the specified name. When the command is executed, this will save the current routing configuration of all MXNet encoders/decoders in the system.

Example command & feedback response:

matrix preset add Preset1

{"code":0,"cmd":"matrix preset add Preset1"}

matrix preset rm {name}

Function

Deletes the matrix preset associated with the name entered.

Example command & feedback response:

Matrix preset rm Preset1

{"cmd":"matrix preset rm Preset1","info":"OK","code":0}

matrix preset active {name} [force]

Function

Activates a preset specified by the {name} command argument. If the argument [force] is added after the {name} argument, the Control box will force a reconnection for all routes and devices affected by the matrix preset.

Example command & feedback response:

matrix preset active Preset1

{"cmd":"matrix preset active Preset1","info":"OK","code":0}

IPC Scene Commands

Note: deprecation, recommend to use command **vwid**.

Use for scene setting.

scene list

Function

Get scene list.

Example

scene list

{"cmd":"scene list","info":{"scene1":{"matrix":["mx1"],"vwall":["vw1"]}},"code":0}

scene get {name}

Function

Get scene {name}'s info.

Example

scene get scene1

{"cmd":"scene get scene1","info":{"matrix":["mx1"],"vwall":["vw1"]},"code":0}

scene set vw {name} {vw1 ...}

Function

Put {vw1} of video wall into {name} of scene..

Example

scene set vw scene1 vw1

{"cmd":"scene set vw scene1 vw1","info":"OK","code":0}

scene set matrix {name} {mtx1 ...}

Function

Put {mtx1} of matrix into scene {name}..

Example

scene set matrix scene1 mx1

{"cmd":"scene set matrix scene1 mx1","info":"OK","code":0}

scene rm {name}

Function

Delete scene {name}.

Example

scene rm scene1

{"cmd":"scene rm scene1","info":"OK","code":0}

scene active {name} [force]

Function

Active scene {name}, if add [force], IPC will force refresh all the devices of scene{name}.

Example

scene active scene1

{"cmd":"scene active scene1","info":"OK","code":0}

IPC vwid Commands

Used for videowall and layout configuration and generation.

vwid list

Function

Get vwid list

Example

vwid list

{"cmd":"vwid list","info":{"vw2":{"cols":3,"rows":3,"layouts":{"s2":{"cols":3,"rows":3,"layout":["1:1:::2:2:1:1:0:2:1:1:1:1","1:2:::2:2:1:2:0:2:1:1:1:1","1:3:::1:1:1:1:0:2:1:1:1:1","2:1:::2:2:2:1:0:2:1:1:1:1","2:2:::2:2:2:2:0:2:1:1:1:1","2:3:::1:1:1:1:0:2:1:1:1:1","3:1:::1:1:1:1:0:2:1:1:1:1","3:2:::1:1:1:1:0:2:1:1:1:1","3:3:::1:1:1:1:0:2:1:1:1:1"]},"scene1":{"cols":3,"layout":[],"rows":3}}},"videowall1":{"cols":2,"rows":3,"layouts":{"vlayout1":{"cols":2,"rows":3,"layout":["1:1:TX1:RX1:1:1:1:1:0:2:1:1:1:1","1:2:TX1:RX2:1:1:1:1:0:2:1:1:1:1"]},"vlayout1_new":{"cols":2,"rows":3,"layout":["1:1:TX1:RX1:1:1:1:1:0:2:1:1:1:1","1:2:TX1:RX2:1:1:1:1:0:2:1:1:1:1"]}}}},"code":0}

vwid get {videowall_name}

Function

Get vwid {videowall_name}'s info

Example

vwid get videowall1

{"cmd":"vwid get videowall1","info":{"cols":2,"rows":3,"layouts":{"vlayout1":{"cols":2,"rows":3,"layout":["1:1:TX1:RX1:1:1:1:1:0:2:1:1:1:1","1:2:TX1:RX2:1:1:1:1:0:2:1:1:1:1"]},"vlayout1_new":{"cols":2,"rows":3,"layout":["1:1:TX1:RX1:1:1:1:1:0:2:1:1:1:1","1:2:TX1:RX2:1:1:1:1:0:2:1:1:1:1"]}}},"code":0}

vwid rm {videowall_name}

Function

Delete vwid {videowall_name}

Example

vwid rm videowall1

{"cmd":"vwid rm videowall1","info":"OK","code":0}

vwid add {videowall_name} {rows} {cols}

Function

Add vwid, name is videowall_name, total rows is {rows}, total columns is {cols}. **Note: videowall_name can not contain a colon ':'.**

Example

vwid add videowall1 2 2

{"cmd":"vwid add videowall1 2 2","info":"OK","code":0}

vwid setrowcol {videowall_name} {rows} {cols}

Function

Set {rows} and {cols} of {videowall_name}

Example

vwid setrowcol videowall1 5 5

{"cmd":"vwid setrowcol videowall1 5 5","info":"OK","code":0}

vwid layout add {videowall_name} {layout_name}

Function

Add a layout on {videowall_name}, its name is {layout_name}. Note: {videowall_name}'s rows and cols value will be automatically copied to {layout_name}'s rows and cols. **Note: videowall_name and layout_name can not contain a colon ':'.**

Example

vwid layout add videowall1 vlayout1

{"cmd":"vwid layout add videowall1 vlayout1","info":"OK","code":0}

vwid layout setrowcol {videowall_name} {layout_name} {rows} {cols}

Function

Set layout {layout_name}’s row and column to {rows} and {cols} of {videowall_name}. Note: {videowall_name}’s rows and columns will be automatically modified to {rows} and {cols}.

Example

vwid layout setrowcol videowall1 vlayout1 3 2

{"cmd":"vwid layout setrowcol videowall1 vlayout1 3 2","info":"OK","code":0}

vwid layout list {videowall_name}

Function

Get layout list of vwid

Example

vwid layout list videowall1

{"cmd":"vwid layout list videowall1","info":{"vlayout1":{"cols":2,"rows":3,"layout":["1:1:TX1:RX1:1:1:1:1:0:2:1:1:1:1","1:2:TX1:RX2:1:1:1:1:0:2:1:1:1:1"]},"vlayout1_new":{"cols":2,"rows":3,"layout":["1:1:TX1:RX1:1:1:1:1:0:2:1:1:1:1","1:2:TX1:RX2:1:1:1:1:0:2:1:1:1:1"]}},"code":0}

vwid layout get {videowall_name} {layout_name}

Function

Get {layout_name}’s info of {videowall_name}.

Example

vwid layout get videowall1 vlayout1

{"cmd":"vwid layout get videowall1 vlayout1","info":{"cols":2,"rows":3,"layout":["1:1:TX1:RX1:1:1:1:1:0:2:1:1:1:1","1:2:TX1:RX2:1:1:1:1:0:2:1:1:1:1"]},"code":0}

vwid layout set {videowall_name} {layout_name} ...

Format:

vwid layout set {videowall_name} {layout_name} row:col:tx:rx:vwrows:vwcols:vwrow:vwcol:rotate:stretch:vw:ow:vh:oh row:col:tx:rx:vwrows:vwcols:vwrow:vwcol:rotate:stretch:vw:ow:vh:oh ...

Function

Set **row:col:tx:rx:vwrows:vwcols:vwrow:vwcol:rotate:stretch:vw:ow:vh:oh** in {layout_name} of {videowall_name}.

row: the row in the layout, start with 1.

col: the column in the layout, start with 1.

tx: the encoder device id or mac.

rx: the decoder device id or mac.

vwrows: in internal video wall total rows.

vwcols: in internal video wall total columns.

vwrow: the device in the row of internal video wall.

vwcol: the device in the column of internal video wall.

Note: When make matrix, vwrows:vwcols:vwrow:vwcol will be 1:1:1:1

rotate: the roata can be 0, 180 or 270.

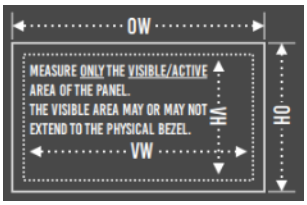
stretch: the stretch can be 1 or 2. 1 is stretch out, 2 is fit in.

vw: visible width of display.

ow: outer width of display.

vh: visible height of display.

oh: outer height of display.



Note: these params join with ":", can connect more than one value, each value is separated by spaces, vw:ow:vh:oh default value is 1:1:1:1.

Example

vwid layout set videowall1 vlayout1 1:1:TX1:RX1:1:1:1:1:0:2:1:1:1:1 1:2:TX1:RX2:1:1:1:1:0:2:1:1:1:1

2:1:TX1:RX3:2:2:1:1:180:2:1:1:1:1 2:2:TX1:RX4:2:2:1:2:180:2 3:1:TX1:RX5:2:2:2:1:0:2:1:1:1:1

{"cmd":"vwid layout set videowall1 vlayout1 1:1:TX1:RX1:1:1:1:1:0:2:1:1:1:1 1:2:TX1:RX2:1:1:1:1:0:2:1:1:1:1

2:1:TX1:RX3:2:2:1:1:180:2:1:1:1:1 2:2:TX1:RX4:2:2:1:2:180:2 3:1:TX1:RX5:2:2:2:1:0:2:1:1:1:1","info":"OK","code":0}

vwid layout tx {videowall_name} {layout_name} {row:column/OldTxID/OldTxMAC/ALL} {NewTxID/NewTxMAC}

Function

In the videowall layout {videowall_name} {layout_name}, set old tx **row:column/OldTxID/OldTxMAC/ALL** to new tx {NewTxID/NewTxMAC}

Note: Top left corner starts at 1:1. Cbox version 2.17 and above support.

Example

vwid layout tx videowall1 layout1 1:2 188A6A8CC3DC

{"cmd":"vwid layout tx videowall1 layout1 1:2 188A6A8CC3DC","info":"","code":0}

vwid layout tx videowall1 layout1 188A6A8CC3DE 188A6A8CC3DC

{"cmd":"vwid layout tx videowall1 layout1 188A6A8CC3DE 188A6A8CC3DC","info":"","code":0}

vwid layout tx videowall1 layout1 TX1 TX2

{"cmd":"vwid layout tx videowall1 layout1 TX1 TX2","info":"","code":0}

vwid layout tx videowall1 layout1 TX3 188A6A8CC3DC

{"cmd":"vwid layout tx videowall1 layout1 TX3 188A6A8CC3DC","info":"","code":0}

vwid layout tx videowall1 layout1 ALL 188A6A8CC3DC

{"cmd":"vwid layout tx videowall1 layout1 ALL 188A6A8CC3DC","info":"","code":0}

vwid layout osd {videowall_name} {layout_name} {on/off}

Function

Show or hide osd on {layout_name} of {videowall_name}

Example

vwid layout osd videowall1 vlayout1 on

{"cmd":"vwid layout osd videowall1 vlayout1 on","info":"OK","code":0}

or

vwid layout osd videowall1 vlayout1 off

{"cmd":"vwid layout osd videowall1 vlayout1 off","info":"OK","code":0}

vwid layout rm {videowall_name} {layout_name}

Function

Remove {layout_name} of {videowall_name}.

Example

vwid layout rm videowall1 vlayout1

{"cmd":"vwid layout rm videowall1 vlayout1","info":"OK","code":0}

vwid layout copy {videowall_name} {layout_name} {new_layout_name}

Function

Copy {layout_name} of {videowall_name} to new layout named {new_layout_name}.

Example

vwid layout copy videowall1 vlayout1 vlayout1_new

{"cmd":"vwid layout copy videowall1 vlayout1 vlayout1_new","info":"OK","code":0}

vwid layout active {videowall_name} {layout_name}

Function

Active {layout_name} of {videowall_name}.

Example

vwid layout active videowall1 vlayout1

{"cmd":"vwid layout active videowall1 vlayout1","info":"OK","code":0}

vwid layout combine {videowall_name} {layout_name} {leftTopRX_ID or row:col} {vwrow} {vwcol}

Function

Combine {layout_name} of {videowall_name} from left top RX or left top RX’s position row and col to VIDEO WALL(vwrow and vwcol).

Example

vwid layout combine videowall1 vlayout1 1:2 1 2

{"cmd":"vwid layout combine videowall1 vlayout1 1:2 1 2","info":"OK","code":0}

or

vwid layout combine videowall1 vlayout1 RX1 1 2

{"cmd":"vwid layout combine videowall1 vlayout1 RX1 1 2","info":"OK","code":0}

vwid layout split {videowall_name} {layout_name} {RX_ID or row:col}

Function

Split {layout_name} of {videowall_name} RX_ID(or row:col) to MATRIX.

Example

vwid layout split videowall1 vlayout1 1:2

{"cmd":"vwid layout split videowall1 vlayout1 1:2","info":"OK","code":0}

or

vwid layout split videowall1 vlayout1 RX1

{"cmd":"vwid layout split videowall1 vlayout1 RX1","info":"OK","code":0}

IPC mosaic Commands

Use for mosaic videowall and layout setting.

vw mosaic list

Function

Get mosaic list

Example

vw mosaic list

{"cmd":"vw mosaic list","info":{"videowall1":{"cols":2,"rows":2,"layouts":{"layout1":{"layout":[{"captureLeftOffset":"0","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":1,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"0"},{"captureLeftOffset":"960","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":2,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"0"},{"captureLeftOffset":"0","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":3,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"541"},{"captureLeftOffset":"960","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":4,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"541"}]}},"displays":[],"totalOutputs":3}},"code":0}

vw mosaic get {videowall_name}

Function

Get mosaic videowall {videowall_name}'s info

Example

vw mosaic get videowall1

{"cmd":"vw mosaic get videowall1","info":{"cols":2,"rows":2,"layouts":{"layout1":{"layout":[{"captureLeftOffset":"0","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":1,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"0"},{"captureLeftOffset":"960","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":2,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"0"},{"captureLeftOffset":"0","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":3,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"541"},{"captureLeftOffset":"960","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":4,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"541"}]}},"displays":[],"totalOutputs":3},"code":0}

vw mosaic rm {videowall_name}

Function

Delete mosaic videowall {videowall_name}

Example

vw mosaic rm videowall1

{"cmd":"vw mosaic rm videowall1","info":"OK","code":0}

vw mosaic add {videowall_name} {rows} {cols} {total_outputs}

Function

Add or modify mosaic videowall, name is videowall_name, total rows is {rows}, total columns is {cols}, actual total outputs is {total_outputs}. **Note: videowall_name can not contain a colon ':'.**

Example

vw mosaic add videowall1 2 2 3

{"cmd":"vw mosaic add videowall1 2 2 3","info":"OK","code":0}

vw mosaic layout add {videowall_name} {layout_name}

Function

Add a layout on {videowall_name}, its name is {layout_name}. **Note: videowall_name and layout_name can not contain a colon ':'.**

Example

vw mosaic layout add videowall1 layout1

{"cmd":"vw mosaic layout add videowall1 layout1","info":"OK","code":0}

vw mosaic layout list {videowall_name}

Function

Get layout list of mosaic videowall

Example

vw mosaic layout list videowall1

{"cmd":"vw mosaic layout list videowall1","info":{"layout1":{"layout":[{"captureLeftOffset":"0","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":1,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"0"},{"captureLeftOffset":"960","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":2,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"0"},{"captureLeftOffset":"0","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":3,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"541"},{"captureLeftOffset":"960","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":4,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"541"}]}},"code":0}

vw mosaic layout get {videowall_name} {layout_name}

Function

Get {layout_name}'s info of {videowall_name}.

Example

vw mosaic layout get videowall1 layout1

{"cmd":"vw mosaic layout get videowall1 layout1","info":{"layout":[{"captureLeftOffset":"0","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":1,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"0"},{"captureLeftOffset":"960","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":2,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"0"},{"captureLeftOffset":"0","canvasHeight":"1080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":3,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"541"},{"captureLeftOffset":"960","canvasHeight":"1

080","client":"188A6A45C4A5","host":"188A6A8CC3DC","rotation":"0","canvasWidth":"1920","index":4,"captureWidth":"960","captureHeight":"541","resolution":5,"captureTopOffset":"541"}]},"code":0}

vw mosaic layout rm {videowall_name} {layout_name}

Function

Remove {layout_name} of {videowall_name}.

Example

vw mosaic layout rm videowall1 layout1

{"cmd":"vw mosaic layout rm videowall1 layout1","info":"OK","code":0}

vw mosaic layout canvas {videowall_name} {layout_name} {indexid} {canvasWidth} {canvasHeight}

Function

Set the source canvas's canvasWidth and canvasHeight of {layout_name}. Indexid's value is from output1 to {total_outputs}.

Note: Default canvasWidth is 1920, and default canvasHeight is 1080.

Example

vw mosaic layout canvas videowall1 layout1 1 1920 1080

{"cmd":"vw mosaic layout canvas videowall1 layout1 1 1920 1080","info":"OK","code":0}

vw mosaic layout canvas videowall1 layout1 2 1920 1080

{"cmd":"vw mosaic layout canvas videowall1 layout1 2 1920 1080","info":"OK","code":0}

vw mosaic layout canvas videowall1 layout1 3 1920 1080

{"cmd":"vw mosaic layout canvas videowall1 layout1 3 1920 1080","info":"OK","code":0}

vw mosaic layout canvas videowall1 layout1 4 1920 1080

{"cmd":"vw mosaic layout canvas videowall1 layout1 4 1920 1080","info":"OK","code":0}

vw mosaic layout capture {videowall_name} {layout_name} {indexid} {captureWidth} {captureHeight} {captureLeftOffset} {captureTopOffset}

Function

In the mosaic layout {videowall_name} {layout_name}, capture the width{captureWidth} and height{captureHeight} of the output {indexid}, and the offset of the upper left corner from the origin is {captureLeftOffset} and {captureTopOffset}.

Example

//left top

vw mosaic layout capture videowall1 layout1 1 960 541 0 0

{"cmd":"vw mosaic layout capture videowall1 layout1 1 960 541 0 0","info":"OK","code":0}

//right top

vw mosaic layout capture videowall1 layout1 2 960 541 960 0

{"cmd":"vw mosaic layout capture videowall1 layout1 2 960 541 960 0","info":"OK","code":0}

//left bottom

vw mosaic layout capture videowall1 layout1 3 960 541 0 541

{"cmd":"vw mosaic layout capture videowall1 layout1 3 960 541 0 541","info":"OK","code":0}

//right bottom

vw mosaic layout capture videowall1 layout1 4 960 541 960 541

{"cmd":"vw mosaic layout capture videowall1 layout1 4 960 541 960 541","info":"OK","code":0}

vw mosaic layout resolution {videowall_name} {layout_name} {indexid} {resolution}

Function

In the mosaic layout {videowall_name} {layout_name}, set {resolution} of the output {indexid}.

The {resolution} is [0~8], 0:passthrough, 1:720P50, 2:720P60, 3:1080P24, 4:1080P50, 5:1080P60, 6:4K30, 7:4K50, 8:4K60.

Note: 0:passthrough is only for device AC-MXNET-1G-R.

Example

vw mosaic layout resolution videowall1 layout1 1 5

{"cmd":"vw mosaic layout resolution videowall1 layout1 1 5","info":"OK","code":0}

vw mosaic layout resolution videowall1 layout1 2 5

{"cmd":"vw mosaic layout resolution videowall1 layout1 2 5","info":"OK","code":0}

vw mosaic layout resolution videowall1 layout1 3 5

{"cmd":"vw mosaic layout resolution videowall1 layout1 3 5","info":"OK","code":0}

vw mosaic layout resolution videowall1 layout1 4 5

{"cmd":"vw mosaic layout resolution videowall1 layout1 4 5","info":"OK","code":0}

vw mosaic layout rotation {videowall_name} {layout_name} {indexid} {rotation}

Function

In the mosaic layout {videowall_name} {layout_name}, set {rotation} of the output {indexid}.

The {rotation} is [0,180,270].

Note: Rotation 180 and 270 are only for device AC-MXNET-1G-R.

Example

vw mosaic layout rotation videowall1 layout1 1 0

{"cmd":"vw mosaic layout rotation videowall1 layout1 1 0","info":"OK","code":0}

vw mosaic layout rotation videowall1 layout1 2 0

{"cmd":"vw mosaic layout rotation videowall1 layout1 2 0","info":"OK","code":0}

vw mosaic layout rotation videowall1 layout1 3 0

{"cmd":"vw mosaic layout rotation videowall1 layout1 3 0","info":"OK","code":0}

vw mosaic layout rotation videowall1 layout1 4 0

{"cmd":"vw mosaic layout rotation videowall1 layout1 4 0","info":"OK","code":0}

vw mosaic layout tx {videowall_name} {layout_name} {indexid} {TxID/TxMAC}

Function

In the mosaic layout {videowall_name} {layout_name}, set tx {TxID/TxMAC} of the output {indexid}.

Example

vw mosaic layout tx videowall1 layout1 1 188A6A8CC3DC

{"cmd":"vw mosaic layout tx videowall1 layout1 1 188A6A8CC3DC","info":"OK","code":0}

vw mosaic layout tx videowall1 layout1 2 188A6A8CC3DC

{"cmd":"vw mosaic layout tx videowall1 layout1 2 188A6A8CC3DC","info":"OK","code":0}

vw mosaic layout tx videowall1 layout1 3 188A6A8CC3DC

{"cmd":"vw mosaic layout tx videowall1 layout1 3 188A6A8CC3DC","info":"OK","code":0}

vw mosaic layout tx videowall1 layout1 4 188A6A8CC3DC

{"cmd":"vw mosaic layout tx videowall1 layout1 4 188A6A8CC3DC","info":"OK","code":0}

vw mosaic layout rx {videowall_name} {layout_name} {indexid} {TxID/TxMAC}

Function

In the mosaic layout {videowall_name} {layout_name}, set rx {RxID/RxMAC} of the output {indexid}.

Example

vw mosaic layout rx videowall1 layout1 1 188A6A45C4A5

{"cmd":"vw mosaic layout rx videowall1 layout1 1 188A6A45C4A5","info":"OK","code":0}

vw mosaic layout rx videowall1 layout1 2 188A6A45C4A6

{"cmd":"vw mosaic layout rx videowall1 layout1 2 188A6A45C4A6","info":"OK","code":0}

vw mosaic layout rx videowall1 layout1 3 188A6A45C4A7

{"cmd":"vw mosaic layout rx videowall1 layout1 3 188A6A45C4A7","info":"OK","code":0}

vw mosaic layout rx videowall1 layout1 4 188A6A45C4A8

{"cmd":"vw mosaic layout rx videowall1 layout1 4 188A6A45C4A8","info":"OK","code":0}

vw mosaic layout osd {videowall_name} {layout_name} {on/off} {on/off}

Function

Show or hide osd on {layout_name} of {videowall_name}

Example

vw mosaic layout osd videowall1 layout1 on

{"cmd":"vw mosaic layout osd videowall1 layout1 on","info":"OK","code":0}

or

vw mosaic layout osd videowall1 layout1 off

{"cmd":"vw mosaic layout osd videowall1 layout1 off","info":"OK","code":0}

vw mosaic layout copy {videowall_name} {layout_name} {new_layout_name}

Function

Copy {layout_name} of {videowall_name} to new layout named {new_layout_name}.

Example

vw mosaic layout copy videowall1 layout1 layout1_new

{"cmd":"vw mosaic layout copy videowall1 layout1 layout1_new","info":"OK","code":0}

vw mosaic layout active {videowall_name} {layout_name}

Function

Active {layout_name} of {videowall_name}.

Example

vw mosaic layout active videowall1 layout1

{"cmd":"vw mosaic layout active videowall1 layout 1","info":"OK","code":0}

IPC kvm Commands

Use for kvm roaming.

kvm list

Function

Get kvm list

Example

kvm list

{"cmd":"kvm list","info":{"kvm1":{"cols":2,"rows":3,"layout":["1:1:tx1:rx1:0","1:2:tx2:rx2:0","2:1:tx3:rx3:0","2:2:tx4:rx4:1","3:1:tx5:rx5:0","3:2:tx6:rx6:0"]}},"code":0}

kvm get {kvm_name}

Function

Get info of {kvm_name}

Example

kvm get kvm1

{"cmd":"kvm get kvm1","info":{"cols":2,"rows":3,"layout":["1:1:tx1:rx1:0","1:2:tx2:rx2:0","2:1:tx3:rx3:0","2:2:tx4:rx4:1","3:1:tx5:rx5:0","3:2:tx6:rx6:0"]},"code":0}

kvm add {kvm_name} {rows} {cols}

Function

Add {kvm_name}, its row and col is {rows} and {cols}.

Example

kvm add kvm1 3 2

{"cmd":"kvm add kvm1 3 2","info":"OK","code":0}

kvm rm {kvm_name}

Function

Romove kvm {kvm_name}.

Example

kvm rm kvm1

{"cmd":"kvm rm kvm1","info":"OK","code":0}

kvm set {kvm_name} row:col:tx:rx:isprimary row:col:tx:rxisprimary ...

Function

Set kvm {kvm_name}'s info in row and col.

row: The row in kvm's rows.

col: The col in kvm's cols.

tx: encoder.

rx: decoder.

isprimary: Its value is 1 or 0. 1 means the rx is primary, 0 meas the rx is secondary. **Note, only one primary in one kvm.**

Example

kvm set kvm1 1:1:tx1:rx1:0 1:2:tx2:rx2:0 2:1:tx3:rx3:0 2:2:tx4:rx4:1 3:1:tx5:rx5:0 3:2:tx6:rx6:0

{"cmd":"kvm set kvm1 1:1:tx1:rx1:0 1:2:tx2:rx2:0 2:1:tx3:rx3:0 2:2:tx4:rx4:1 3:1:tx5:rx5:0 3:2:tx6:rx6:0","info":"OK","code":0}

kvm osd {kvm_name} {on/off}

Function

Set {kvm_name}'s osd on or off.

Example

kvm osd kvm1 on

{"cmd":"kvm osd kvm1 on","info":"OK","code":0}

or

kvm osd kvm1 off

{"cmd":"kvm osd kvm1 off","info":"OK","code":0}

kvm setrowcol {kvm_name} {rows} {cols}

Function

Modify {kvm_name}'s rows and cols to {rows} and {cols}.

Example

kvm setrowcol kvm1 3 2

{"cmd":"kvm setrowcol kvm1 3 2","info":"OK","code":0}

kvm active {kvm_name}

Function

Active {kvm_name}. **Note, this command will make decoder's video, audio, usb be switched to corresponding encoder.**

Example

kvm active kvm1

{"cmd":"kvm active kvm1","info":"OK","code":0}