IPLinx Programming Guide

Rev 160831

11675 Ridgeline Dr.
Colorado Springs, CO 80921

Toll-Free: 800-530-8998
Fax: 719-260-0075
supportlibav@libav.com

Table of Contents

API Overview

The API command feedback supports alias names for system control and management. In order to use the alias names, transmit the command `config set session alias on`. After the command is sent, all transmissions and responses will use the aliases of the encoders and decoders. If the IPEXCB is powered off or restarted, the command will need to be retransmitted to properly interact with the alias names of the encoders and decoders again.

LAN Connections

The IPEXCB has two Ethernet ports: LAN1 (AV/PoE) and LAN2 (CTRL). It listens to TCP port 23 on both ports, through which you can control and manage the IP video matrix with the API commands in this document.

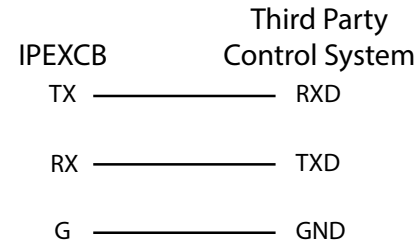
LAN1 (AV/PoE) is used to connect the IPEXCB to the Ethernet switch. The default IP address for LAN1 of the IPEXCB is 169.254.1.1.

LAN2 (CTRL) is used to connect the IPEXCB to a third party control system. The default IP address for LAN2 of the IPEXCB is 192.168.11.243.

RS232 Connections

The IPEXCB features two RS232 connections: Debug and Control. The Debug connection will only communicate with the IPEXCB and will not control any encoders or decoders.

To use the RS232 control transport capabilities of the IPEXCB, connect the TX, RX, and ground control signal wires to the middle RS232 connections on the removable 6-pole terminal block. Consult the manual of the control device to determine which pins the TX and RX signals are carried on. Be sure to always connect TX to RX and RX to TX.



The RS232 control ports require a standard straight-through serial cable for operation. The default settings for the RS232 ports are:

- Debug connection: 115200 baud, 8 Data Bits, 1 Stop Bit, Parity = none
- Control connection: 9600 baud, 8 Data Bits, 1 Stop Bit, Parity = none

While the IPEXCB requires RS232 commands to be sent to it at 9600 baud through the control connection, multiple baud rates are available to communicate with the remote devices.

Terminology

Below is a list of the terminology used in this document with a description of its use.

Terminology	Description
Device	Encoder or decoder
Online	Device is working properly and can be controlled by the IPEXCB
Offline	Device cannot be controlled by the IPEXCB for some reason, such as loss of power or disconnected from the Ethernet switch
Device Name	Default name of the device (device type-MAC address), such as IPEX2002-341B22FFFFB3
Alias	Name assigned to device for easy management; only alphanumeric characters and hyphens (-) are supported.

API Commands

Below is a list of the primary commands used in this document with a description of its use.

Commands	Description
`config`	Manage and configure IPEXCB and devices
`matrix`	Control the switching of devices
`vw`	Manage and configure video wall applications
`serial`	Transmit commands to sources and displays via serial ports on devices
`scene`	Manage and configure video wall layouts
`notify`	Inform third party control device of serial response and online status

Below is a list of the secondary commands used in this document with a description of its use.

Commands	Description
`set`	Applies a change to a primary command
`get`	Queries the active state via primary command

config Commands

config set Commands Summary

Commands	Description
`config set ip4addr`	Configures network settings for LAN1 (AV/PoE)
`config set ip4addr2`	Configures network settings for LAN2 (CTRL)
`config set webloginpasswd`	Changes web GUI login password
`config set restorefactory`	Resets IPEXCB to factory defaults
`config set reboot`	Reboots IPEXCB
`config set device alias`	Renames a device
`config set device remove`	Removes a device record from IPEXCB
`config set device ip`	Configures device network settings
`config set device reboot`	Reboots a device
`config set device restorefactory`	Resets a device to factory defaults
`config set device info`	Changes device working parameters
`config set device cec standby`	Display devices connected to decoder enter standby via CEC
`config set device cec onetouchplay`	Display devices connected to decoder exit standby via CEC
`config set device sinkpower {on|off} hostname1 hostname2 ...`	Multiple display devices connected to decoder enter or exit standby
`config set session alias {on|off}`	Enter or exit the alias mode in current session

config set ip4addr

`config set ip4addr` configures network settings for LAN1 (AV/PoE). This command is used to set the IP address, subnet mask and gateway for the LAN1 (AV/PoE) port. This port only supports static IP mode.

The IPEXCB will automatically reboot after the response for the settings to take effect.

Command Structure

`config set ip4addr` *`xx.xx.xx.xx`* `netmask` *`xx.xx.xx.xx`* `gateway` *`xx.xx.xx.xx`*

Response

`ip setting will change to: ipaddr` *`xx.xx.xx.xx`* `netmask` *`xx.xx.xx.xx`* `gateway` *`xx.xx.xx.xx`*

Example

Set the LAN1 (AV/PoE) port's IP address to 169.254.1.254, subnet mask to 255.255.0.0, and gateway to 169.254.1.1.

Command

```
config set ip4addr 169.254.1.254 netmask 255.255.0.0 gateway 169.254.1.1
```

Response

```
ip setting will change to: ipaddr 169.254.1.254 netmask 255.255.0.0 gateway 169.254.1.1
```

config set ip4addr2

`config set ip4addr2` configures network settings for LAN2 (CTRL). This command is used to set the IP address, subnet mask and gateway for the LAN2 (CTRL) port. This port only supports static IP mode.

The IPEXCB will automatically reboot after the response for the settings to take effect.

Command Structure

`config set ip4addr2` *`xx.xx.xx.xx`* `netmask` *`xx.xx.xx.xx`* `gateway` *`xx.xx.xx.xx`*

Response

`ip setting2 will change to: ipaddr2` *`xx.xx.xx.xx`* `netmask` *`xx.xx.xx.xx`* `gateway` *`xx.xx.xx.xx`*

Example

Set the LAN2 (CTRL) port's IP address to 192.168.11.243, subnet mask to 255.255.255.0, and gateway to 192.168.11.1.

Command

```
config set ip4addr2 192.168.11.243 netmask 255.255.0.0 gateway 192.168.11.1
```

Response

```
ip setting2 will change to: ipaddr 192.168.11.243 netmask 255.255.0.0 gateway 192.168.11.1
```

config set webloginpasswd

`config set webloginpasswd` changes the web GUI login password.

Command Structure

`config set webloginpasswd xxxxxx`

Response

`password for web modified`

Example

Change the web GUI login password to 123456.

Command

`config set webloginpasswd 123456`

Response

`password for web modified`

config set restorefactory

`config set restorefactory` resets the IPEXCB to factory defaults.

The IPEXCB will automatically reboot after the response for the settings to take effect.

Command

`config set restorefactory`

Response

`system will restore to factory settings now`

config set reboot

`config set reboot` reboots the IPEXCB.

Command

`config set reboot`

Response

`system will reboot now`

config set device alias

`config set device alias` renames a device for easier identification and management. An alias can be used in any other command to replace the device name. Each alias should be unique. Only alphanumeric characters and hyphens (-) may be used in the device name.

`hostname` is the device's default name.

Command Structure

`config set device alias` *`hostname xxxx`*

Response

`hostname``'s alias is` *`xxxx`*

Example

Rename IPEX2002-341B22FFFFB3 to DVD1.

Command

```
config set device alias IPEX2002-341B22FFFFB3 MYDVD
```

Response

```
IPEX2002-341B22FFFFB3's alias is MYDVD
```

config set device remove

`config set device remove` removes a device record from the IPEXCB. One or more device records may be removed at the same time. When a device's record is removed, it cannot be detected or controlled by the IPEXCB. To restore the removed online device, reboot it or the IPEXCB.

`hostname1` and *`hostname2`* are device names, which may be the default name or an alias.

Command Structure

`config set device remove` *`hostname1 hostname2`*`...`

Response

```
The following device's record will be removed:
hostname1
hostname2
...
```

Example

Remove the records of IPEX2001-AABBCCEEDDFF and IPEX2002-1234567890AB.

Command

```
config set device remove IPEX2001-AABBCCEEDDFF IPEX2002-1234567890AB
```

Response

```
the following device's record will be removed:
IPEX2001-AABBCCEEDDFF
IPEX2002-1234567890AB
```

config set device ip

`config set device ip` configures device network settings. Devices support AutoIP, DHCP and Static IP for network configuration. For Static IP, the IP address, subnet mask and gateway must be declared at the same time. One or more devices may be configured at the same time. The device will need to be manually rebooted for the settings to take effect.

hostname1 and *hostname2* are device names, which may be the default name or an alias.

Command Structure

```
config set device ip hostname1 {autoip|dhcp|static ip4addr netmask gateway}, hostname2
{autoip|dhcp|static ip4addr netmask gateway}...
```

Response

```
Devices’ ipsetting will change to:
hostname1 {autoip|dhcp|static ip4addr netmask gateway}
hostname2 {autoip|dhcp|static ip4addr netmask gateway}
...
```

Example

Set IPEX5002-341B22800BCD to AutoIP and IPEX5002-341B22800BCA to Static IP (IP address 169.254.5.253, subnet mask 255.255.0.0, gateway 169.254.1.253).

Command

```
config set device ip IPEX5002-341B22800BCD autoip, IPEX5002-341B22800BCA static
169.254.5.253 255.255.0.0 169.254.1.253
```

Response

```
Devices's ipsetting will change to:
IPEX5002-341B22800BCD autoip
IPEX5002-341B22800BCA static 169.254.5.253 255.255.0.0 169.254.1.253
```

config set device reboot

`config set device reboot` reboots one or more devices.

`hostname1` and *`hostname2`* are device names, which may be the default name or an alias.

Command Structure

```
config set device reboot hostname1 hostname2...
```

Response

```
the following device will reboot now:
hostname1
hostname2
...
```

Example

Reboot IPEX2001-341B22FFFFB3 and IPEX2002-341B22FFFFB4.

Command

```
config set device reboot IPEX2001-341B22FFFFB3 IPEX2002-341B22FFFFB4
```

Response

```
the following device will reboot now:
IPEX2001-341B22FFFFB3
IPEX2002-341B22FFFFB4
```

config set device reboot

`config set device restorefactory` resets one or more devices to the factory default settings. The device will automatically reboot after the response for the settings to take effect.

`hostname1` and *`hostname2`* are device names, which may be the default name or an alias.

Command Structure

```
config set device restorefactory hostname1 hostname2...
```

Response

```
the following device will restore to factory setting now:
hostname1
hostname2
...
```

Example

Reset IPEX2001-341B22FFFFB3 and IPEX2002-341B22FFFFB4 to factory defaults.

Command

```
config set device restore IPEX2001-341B22FFFFB3 IPEX2002-341B22FFFFB4
```

Response

```
the following device will restore to factory setting now:
IPEX2001-341B22FFFFB3
IPEX2002-341B22FFFFB4
```

config set device info

`config set device info` changes device working parameters, such as bandwidth utilization or IP settings. One or more devices plus one or more settings may be configured at the same time. The device will need to be manually rebooted for the settings to take effect.

`hostname1` and *`hostname2`* are device names, which may be the default name or an alias. *`key`* is the parameter, and *`value`* is the value. Please see page 37 for the keys and acceptable values.

Command Structure

```
config set device info key1=value1 [key2=value2...] hostname1 hostname2...
```

Response

```
config set device info key1=value1 key2=value2 key3=value3 key4=value4 hostname1
hostname2...
```

Example

Set IPEX5002-341B22800BCD and IPEX5002-341B22800BCA to AutoIP.

Command

```
config set device info ip_mode=autoip IPEX5002-341B22800BCA IPEX5002-341B22800BCD
```

Response

```
config set device info ip_mode=autoip IPEX5002-341B22800BCA IPEX5002-341B22800BCD
```

config set device cec standby

`config set device cec standby` transmits the CEC enter standby command (power off) to a CEC capable display device connected to a specific decoder. One or more devices may be configured at the same time.

`hostname1` and *`hostname2`* are device names, which may be the default name or an alias.

Command Structure

```
config set device cec standby hostname1 hostname2...
```

Response

```
config set device cec standby hostname1 hostname2...
```

Example

Power off the display connected to IPEX2002-341B22FFFFB4 via CEC.

Command

```
config set device cec standby IPEX2002-341B22FFFFB4
```

Response

```
config set device cec standby IPEX2002-341B22FFFFB4
```

config set device cec onetouchplay

`config set device cec onetouchplay` transmits the CEC exit standby command (power on) to a CEC capable display device connected to a specific decoder. One or more devices may be configured at the same time.

`hostname1` and *`hostname2`* are device names, which may be the default name or an alias.

Command Structure

```
config set device cec onetouchplay hostname1 hostname2...
```

Response

```
config set device cec onetouchplay hostname1 hostname2...
```

Example

Power on the display connected to IPEX2002-341B22FFFFB4 via CEC.

Command

```
config set device cec onetouchplay IPEX2002-341B22FFFFB4
```

Response

```
config set device cec onetouchplay IPEX2002-341B22FFFFB4
```

config set device sinkpower

`config set device sinkpower` transmits the CEC enter standby command (power off) or exit standby command (power off) to a CEC capable display device connected to one or multiple decoders.

`hostname1` and *`hostname2`* are device names, which may be the default name or an alias.

Command Structure

`config set device sinkpower {on|off} hostname1 hostname2...`

Response

`config set device sinkpower {on|off} hostname1 hostname2...`

Example

Power on the displays connected to IPEX5002-341B22800BCD and IPEX5002-341B22800BCA via CEC.

Command

```
config set device sinkpower on IPEX5002-341B22800BCA IPEX5002-341B22800BCD
```

Response

```
config set device sinkpower on IPEX5002-341B22800BCA IPEX5002-341B22800BCD
```

config set session alias

`config set session alias` turns on or off alias mode for the current session. If the value is set to *`on`*, then all API commands will respond with the device alias. If the value is set to *`off`*, then all API commands will respond with the device name.

Command Structure

`config set session alias {on|off}`

Response

`config set session alias {on|off}`

config get Commands Summary

Commands	Description
`config get version`	Queries IPEXCB version information
`config get devicelist`	Queries an online device list
`config get ipsetting`	Queries network settings for LAN1 (AV/PoE)
`config get ipsetting2`	Queries network settings for LAN2 (CTRL)
`config get name`	Queries a device name or its alias
`config get device info`	Queries device working parameters
`config get devicejsonstring`	Queries all device information
`config get scenejsonstring`	Queries all scene information

config get version

`config get version` queries the IPEXCB for the current API, web console and firmware revisions.

Command Structure

```
config get version
```

Response

```
API version: v#.#
System version: LAVS 1.# v#.#.#(v#.#.#)
```

Example

Query the IPEXCB for the current version information.

Command

```
config get version
```

Response

```
API version: v1.9
System version: LAVS 1.0 v7.2.0(v7.2.1)
```

config get devicelist

`config get devicelist` queries the network for a list of all active, online devices. The devices that are returned are all decoders followed by all encoders, but there is no separation between the devices. To get a list of all devices, including offline ones, please use `config get devicejsonstring`.

`hostname1` and *`hostname2`* are device names, which may be the default name or an alias.

Command Structure

```
config get devicelist
```

Response

```
devicelist is hostname1 hostname2...
```

Example

Query the network for active devices after alias naming is turned on.

Command

```
config get devicelist
```

Response

```
devicelist is BottomLeft-2x2 TopRight-2x2 Entrance TopLeft-2x2 BottomRight-2x2 iPad PC
```

config get ipsetting

`config get ipsetting` queries the network settings for LAN1 (AV/PoE).

Command Structure

```
config get ipsetting
```

Response

`ipsetting is:ip4addr` *`xx.xx.xx.xx`* `netmask` *`xx.xx.xx.xx`* `gateway` *`xx.xx.xx.xx`*

Example

Query the network settings for LAN1 (AV/PoE).

Command

```
config get ipsetting
```

Response

```
ipsetting is:ip4addr 169.254.1.1 netmask 255.255.0.0 gateway 169.254.1.254
```

config get ipsetting2

`config get ipsetting2` queries the network settings for LAN2 (CTRL).

Command Structure

`config get ipsetting2`

Response

`ipsetting2 is:ip4addr xx.xx.xx.xx netmask xx.xx.xx.xx gateway xx.xx.xx.xx`

Example

Query the network settings for LAN2 (CTRL).

Command

`config get ipsetting2`

Response

`ipsetting2 is:ip4addr 192.168.11.243 netmask 255.255.0.0 gateway 192.168.11.1`

config get name

`config get name` queries a device name or its alias. A device name can be used to get an alias and vice versa. If `config get name` is used without parameters, the response is all device names and corresponding aliases.

`hostname` is the device name; *`alias`* is the device alias. A device without an alias will respond with *`NULL`*.

Command Structure

`config get name`

Response

`hostname``'s alias is xxxx`

Example 1

Query the alias for IPEX2001-341B22430115.

Command

`config get name IPEX2001-341B22430115`

Response

`IPEX2001-341B22430115's alias is testIPEX`

Example 2

Query the alias for IPEX2001-341B22430225, which is not set.

Command

```
config get name IPEX2001-341B22430225
```

Response

```
IPEX2001-341B22430115's alias is NULL
```

Example 2

Query all device names and their aliases.

Command

```
config get name
```

Response

```
IPEX2002-341B22801761's alias is BottomLeft-2x2
IPEX2002-341B22801763's alias is TopRight-2x2
IPEX2002-341B22801768's alias is TopLeft-2x2
IPEX2002-341B22801769's alias is BottomRight-2x2
IPEX2001-341B22801758's alias is iPad
IPEX2001-341B2280175E's alias is PC
```

config get device info

`config get device info` queries device working parameters, such as bandwidth utilization or IP settings. One or more devices may be queried at the same time.

`hostname1` and *`hostname2`* are device names, which may be the default name or an alias. *`key`* is the parameter, and *`value`* is the value. Please see page 37 for the keys and acceptable values.

Command Structure

```
config get device info hostname1 hostname2...
```

Response

```
devices json info:
{
      "devices":
      [
            {
                  "aliasname" : "xxxx"
                  "key1":"value1"
                  "key2":"value2"
                  ...
            },
            ...
      ]
}
```

Example 1

Query the device info for the device with the alias of `PC`.

Command

```
config get device info PC
```

Response


```
devices json info:
{
      "devices" : [
            {
                  "aliasname" : "PC",
                  "cbr_avg_bitrate" : 10000,
                  "enc_fps" : 60,
                  "enc_gop" : 60,
                  "enc_rc_mode" : "vbr",
                  "fixqp_iqp" : 25,
                  "fixqp_pqp" : 25,
                  "gateway" : "",
                  "hdcp" : true,
                  "ip4addr" : "169.254.135.115",
                  "ip_mode" : "autoip",
                  "mac" : "34:1b:22:80:17:5e",
                  "name" : "IPEX2001-341B2280175E",
                  "netmask" : "255.255.0.0",
                  "profile" : "hp",
                  "sourcein" : "hdmi1",
                  "transport_type" : "raw",
                  "vbr_max_bitrate" : 20000,
                  "vbr_max_qp" : 40,
                  "vbr_min_qp" : 0,
                  "version" : "v2.9.2"
            }
      ]
}
```


Example 2

Query the device info for all devices in the system.

Command

```
config get device info
```

Response


```
devices json info:
{
      "devices" : [
            {
                  "aliasname" : "iPad",
                  "cbr_avg_bitrate" : 10000,
                  "enc_fps" : 60,
                  "enc_gop" : 60,
                  "enc_rc_mode" : "vbr",
                  "fixqp_iqp" : 25,
                  "fixqp_pqp" : 25,
                  "gateway" : "",
                  "hdcp" : true,
                  "ip4addr" : "169.254.197.185",
                  "ip_mode" : "autoip",
                  "mac" : "34:1b:22:80:17:58",
                  "name" : "IPEX2001-341B22801758",
                  "netmask" : "255.255.0.0",
                  "profile" : "hp",
                  "sourcein" : "hdmi1",
                  "transport_type" : "raw",
                  "vbr_max_bitrate" : 20000,
                  "vbr_max_qp" : 40,
                  "vbr_min_qp" : 0,
                  "version" : "v2.9.2"
            },
            {
                  "aliasname" : "PC",
                  "cbr_avg_bitrate" : 10000,
                  "enc_fps" : 60,
                  "enc_gop" : 60,
                  "enc_rc_mode" : "vbr",
                  "fixqp_iqp" : 25,
                  "fixqp_pqp" : 25,
                  "gateway" : "",
                  "hdcp" : true,
                  "ip4addr" : "169.254.135.115",
                  "ip_mode" : "autoip",
                  "mac" : "34:1b:22:80:17:5e",
                  "name" : "IPEX2001-341B2280175E",
                  "netmask" : "255.255.0.0",
                  "profile" : "hp",
                  "sourcein" : "hdmi1",
                  "transport_type" : "raw",
                  "vbr_max_bitrate" : 20000,
                  "vbr_max_qp" : 40,
                  "vbr_min_qp" : 0,
                  "version" : "v2.9.2"
            },
```

```
{
	"aliasname" : "TopLeft-2x2",
	"audio" : [
		{
			"mute" : false,
			"name" : "lineout1"
		}
	],
	"gateway" : "unknown",
	"hdcp" : true,
	"ip4addr" : "169.254.4.68",
	"ip_mode" : "autoip",
	"mac" : "34:1b:22:80:17:68",
	"name" : "IPEX2002-341B22801768",
	"netmask" : "255.255.0.0",
	"sourcein" : "341B2280175E",
	"version" : "v2.9.1"
},
{
	"aliasname" : "TopRight-2x2",
	"audio" : [
		{
			"mute" : false,
			"name" : "lineout1"
		}
	],
	"gateway" : "unknown",
	"hdcp" : true,
	"ip4addr" : "169.254.81.114",
	"ip_mode" : "autoip",
	"mac" : "34:1b:22:80:17:63",
	"name" : "IPEX2002-341B22801763",
	"netmask" : "255.255.0.0",
	"sourcein" : "341B2280175E",
	"version" : "v2.9.1"
},
{
	"aliasname" : "BottomLeft-2x2",
	"audio" : [
		{
			"mute" : false,
			"name" : "lineout1"
		}
	],
	"gateway" : "unknown",
	"hdcp" : true,
	"ip4addr" : "169.254.4.33",
	"ip_mode" : "autoip",
	"mac" : "34:1b:22:80:17:61",
	"name" : "IPEX2002-341B22801761",
	"netmask" : "255.255.0.0",
```

```
                "sourcein" : "341B2280175E",
                "version" : "v2.9.1"
        },
        {
                "aliasname" : "BottomRight-2x2",
                "audio" : [
                        {
                                "mute" : false,
                                "name" : "lineout1"
                        }
                ],
                "gateway" : "unknown",
                "hdcp" : true,
                "ip4addr" : "169.254.4.73",
                "ip_mode" : "autoip",
                "mac" : "34:1b:22:80:17:69",
                "name" : "IPEX2002-341B22801769",
                "netmask" : "255.255.0.0",
                "sourcein" : "341B2280175E",
                "version" : "v2.9.1"
        },
        {
                "aliasname" : "Entrance",
                "audio" : [
                        {
                                "mute" : false,
                                "name" : "lineout1"
                        }
                ],
                "gateway" : "unknown",
                "hdcp" : true,
                "ip4addr" : "169.254.4.53",
                "ip_mode" : "autoip",
                "mac" : "34:1b:22:80:17:65",
                "name" : "IPEX2002-341B22801765",
                "netmask" : "255.255.0.0",
                "sourcein" : "341B2280175E",
                "version" : "v2.9.1"
        }
    ]
}
```

config get devicejsonstring

`config get devicejsonstring` queries all device information for all devices connected to the IPEXCB, including offline devices.

— `aliasname` is the assigned alias for the device; if no alias is shown, the device has not been given an alias.
— `deviceType` is the type of device: encoder (transmitter) or decoder (receiver).
— `group` is the assigned group of the decoder; a decoder can only be in one group.
— `sequence` in a *group* is the position of the group, which starts at 1; if the sequence value is 0, the group is not arranged in a specific order.
— `ip` is the IP address for the device. `online` is the the device activity status: *true* is online, *false* is offline.
— `sequence` is the position of the decoder within a group, which starts at 1; if the sequence value is 0, the device is not arranged in a specific order.
— `trueName` is the device name.

Command Structure

```
config get devicejsonstring
```

Response

```
device json string: [
      {
            "aliasName" : "xxx",
            "deviceType" : "Transmitter/Receiver",
            "group" : [
                  {
                        "name" : "xxx",
                        "sequence" : xxx
                  }
            ],
            "ip" : "xx.xx.xx.xx",
            "online" : true/false,
            "sequence" : xxx,
            "trueName" : "xxx"
      }
      ...
]
```

Example

Query all device information for all devices connected to IPEXCB.

Command

```
config get devicejsonstring
```

Response


```
device json string:[
	{
		"aliasName" : "BottomLeft-2x2",
		"deviceType" : "Receiver",
		"group" : [
			{
				"name" : "2x2-Wall",
				"sequence" : 2
			}
		],
		"ip" : "169.254.4.33",
		"online" : true,
		"sequence" : 4,
		"trueName" : "IPEX2002-341B22801761",
		"txName" : "IPEX2001-341B2280175E"
	},
	{
		"aliasName" : "TopRight-2x2",
		"deviceType" : "Receiver",
		"group" : [
			{
				"name" : "2x2-Wall",
				"sequence" : 2
			}
		],
		"ip" : "169.254.81.114",
		"online" : true,
		"sequence" : 1,
		"trueName" : "IPEX2002-341B22801763",
		"txName" : "IPEX2001-341B2280175E"
	},
	{
		"aliasName" : "Entrance",
		"deviceType" : "Receiver",
		"group" : [
			{
				"name" : "Entrance",
				"sequence" : 3
			}
		],
		"ip" : "169.254.4.53",
		"online" : true,
		"sequence" : 1,
		"trueName" : "IPEX2002-341B22801765",
		"txName" : "IPEX2001-341B2280175E"
	},
	{
		"aliasName" : "TopLeft-2x2",
		"deviceType" : "Receiver",
		"group" : [
			{
				"name" : "2x2-Wall",
				"sequence" : 2
			}
		],
```

```
            "ip" : "169.254.4.68",
            "online" : true,
            "sequence" : 2,
            "trueName" : "IPEX2002-341B22801768",
            "txName" : "IPEX2001-341B2280175E"
      },
      {
            "aliasName" : "BottomRight-2x2",
            "deviceType" : "Receiver",
            "group" : [
                  {
                        "name" : "2x2-Wall",
                        "sequence" : 2
                  }
            ],
            "ip" : "169.254.4.73",
            "online" : true,
            "sequence" : 3,
            "trueName" : "IPEX2002-341B22801769",
            "txName" : "IPEX2001-341B2280175E"
      },
      {
            "aliasName" : "iPad",
            "deviceType" : "Transmitter",
            "group" : [
                  {
                        "name" : "ungrouped",
                        "sequence" : 1
                  }
            ],
            "ip" : "169.254.197.185",
            "online" : true,
            "sequence" : 1,
            "trueName" : "IPEX2001-341B22801758"
      },
      {
            "aliasName" : "PC",
            "deviceType" : "Transmitter",
            "group" : [
                  {
                        "name" : "ungrouped",
                        "sequence" : 1
                  }
            ],
            "ip" : "169.254.135.115",
            "online" : true,
            "sequence" : 2,
            "trueName" : "IPEX2001-341B2280175E"
      }
]
```

config get scenejsonstring

`config get scenejsonstring` queries all scene (layout) information for the IPEXCB.

— `group` defines a group. One scene can only be put in one group.
— `sequence` in *group* defines the position of this group , which starts with 1. If the sequence value is 0, it means that this group is not arranged in specific order.
— `layoutseq` represents the position of the scene in a video wall.
— `n` and `m` define the number of rows and columns respectively in a scene.
— `name` defines the scene name, such as *`s`*.
— `decoderxArray` defines the decoder position within a two-dimensional array in a scene.
— `sequence` in a scene defines the position of a video wall which contains this scene, which starts with 1. If the sequence value is 0, it means that this video wall is not arranged in specific order.
— `encoderxListArray` defines the encoder position within a two-dimensional array in a scene.
— `vwConfigList` defines the configuration of combination screen in a scene.
— `name` defines the combination screen name, which uses the naming structure of *`scene name_combination screen name`* within the IPEXCB.
— `pos_row` defines the start place of the first row.
— `pos_col` defines the start place of the first column.
— `row_count` defines the number of rows in the combination screen.
— `col_count` defines the number of columns in the combination screen.

Command Structure

```
config get scenejsonstring
```

Response

```
scene json string:[
       {
              "group" : [
                     {
                            "name" : "xxx",
                            "sequence" : xxx
                     }
              ],
              "layoutseq" : xxx,
              "m" : xxx,
              "n" : xxx,
              "name" : "xxx-xxx",
              "rxArray" : [
                     [
                            {
                                   "aliasName" : "xxx",
                                   "deviceType" : "Transmitter/Receiver",
                                   "group" : [
                                          {
                                                 "name" : "xxx",
                                                 "sequence" : xxx
                                          }
                                   ],
```

```
                        "online" : true/false,
                        "rxstatus" : xxx,
                        "sequence" : xxx,
                        "trueName" : "xxx",
                        "txName" : "xxx"
                },
                ...
        ],
        [
                {
                        "aliasName" : "xxx",
                        "deviceType" : "Transmitter/Receiver",
                        "group" : [
                                {
                                        "name" : "xxx",
                                        "sequence" : xxx
                                }
                        ],
                        "online" : true/false,
                        "rxstatus" : xxx,
                        "sequence" : xxx,
                        "trueName" : "xxx",
                        "txName" : "xxx"
                },
                ...
        ]
],
"sceneAutoApply" : true/false,
"sequence" : xxx,
"txListArray" : [
        [
                {
                        "devices" : []
                },
                {
                        "devices" : []
                }
        ],
        [
                {
                        "devices" : []
                },
                {
                        "devices" : []
                }
        ]
],
```

```
            "vwConfigList" : [
                  {
                        "col_count" : xxx,
                        "mode" : "xxx",
                        "name" : "xxx_xxx",
                        "oh" : xxx,
                        "ow" : xxx,
                        "pos_col" : xxx,
                        "pos_row" : xxx,
                        "row_count" : xxx,
                        "vh" : xxx,
                        "vw" : xxx
                  },
                  {
                        "col_count" : xxx,
                        "mode" : "xxx",
                        "name" : "xxx_xxx",
                        "oh" : xxx,
                        "ow" : xxx,
                        "pos_col" : xxx,
                        "pos_row" : xxx,
                        "row_count" : xxx,
                        "vh" : xxx,
                        "vw" : xxx
                  }
            ]
      }
]
```

Example

Query all scene information for the IPEXCB, which is configured for a single display plus a two by two video wall.

Command

```
config get scenejsonstring
```

Response

```
scene json string:[
      {
            "group" : [
                  {
                        "name" : "2x2-Wall",
                        "sequence" : 2
                  }
            ],
            "layoutseq" : 1,
            "m" : 2,
            "n" : 2,
            "name" : "Matrix-matrix",
```

```
"rxArray" : [
	[
		{
			"aliasName" : "TopLeft-2x2",
			"deviceType" : "Receiver",
			"group" : [
				{
					"name" : "2x2-Wall",
					"sequence" : 2
				}
			],
			"online" : true,
			"rxstatus" : 1,
			"sequence" : 2,
			"trueName" : "IPEX2002-341B22801768",
			"txName" : "IPEX2001-341B2280175E"
		},
		{
			"aliasName" : "TopRight-2x2",
			"deviceType" : "Receiver",
			"group" : [
				{
					"name" : "2x2-Wall",
					"sequence" : 2
				}
			],
			"online" : true,
			"rxstatus" : 1,
			"sequence" : 1,
			"trueName" : "IPEX2002-341B22801763",
			"txName" : "IPEX2001-341B2280175E"
		}
	],
	[
		{
			"aliasName" : "BottomLeft-2x2",
			"deviceType" : "Receiver",
			"group" : [
				{
					"name" : "2x2-Wall",
					"sequence" : 2
				}
			],
			"online" : true,
			"rxstatus" : 1,
			"sequence" : 4,
			"trueName" : "IPEX2002-341B22801761",
			"txName" : "IPEX2001-341B2280175E"
		},
```

```
                {
                        "aliasName" : "BottomRight-2x2",
                        "deviceType" : "Receiver",
                        "group" : [
                                {
                                        "name" : "2x2-Wall",
                                        "sequence" : 2
                                }
                        ],
                        "online" : true,
                        "rxstatus" : 1,
                        "sequence" : 3,
                        "trueName" : "IPEX2002-341B22801769",
                        "txName" : "IPEX2001-341B2280175E"
                }
        ]
],
"sceneAutoApply" : false,
"sequence" : 1,
"txListArray" : [
        [
                {
                        "devices" : [
                                {
                                        "aliasName" : "iPad",
                                        "deviceType" : "Transmitter",
                                        "group" : [
                                                {
                                                        "name" : "ungrouped",
                                                        "sequence" : 1
                                                }
                                        ],
                                        "online" : true,
                                        "rxstatus" : 1,
                                        "sequence" : 1,
                                        "trueName" : "IPEX2001-341B22801758"
                                }
                        ]
                },
                {
                        "devices" : [
                                {
                                        "aliasName" : "iPad",
                                        "deviceType" : "Transmitter",
                                        "group" : [
                                                {
                                                        "name" : "ungrouped",
                                                        "sequence" : 1
                                                }
                                        ],
                                        "online" : true,
                                        "rxstatus" : 1,
                                        "sequence" : 1,
                                        "trueName" : "IPEX2001-341B22801758"
                                }
                        ]
```

```
                        }
                ],
                [
                        {
                                "devices" : [
                                        {
                                                "aliasName" : "PC",
                                                "deviceType" : "Transmitter",
                                                "group" : [
                                                        {
                                                                "name" : "ungrouped",
                                                                "sequence" : 1
                                                        }
                                                ],
                                                "online" : true,
                                                "rxstatus" : 1,
                                                "sequence" : 2,
                                                "trueName" : "IPEX2001-341B2280175E"
                                        }
                                ]
                        },
                        {
                                "devices" : [
                                        {
                                                "aliasName" : "PC",
                                                "deviceType" : "Transmitter",
                                                "group" : [
                                                        {
                                                                "name" : "ungrouped",
                                                                "sequence" : 1
                                                        }
                                                ],
                                                "online" : true,
                                                "rxstatus" : 1,
                                                "sequence" : 2,
                                                "trueName" : "IPEX2001-341B2280175E"
                                        }
                                ]
                        }
                ]
        ]
},
{
        "group" : [
                {
                        "name" : "2x2-Wall",
                        "sequence" : 2
                }
        ],
        "layoutseq" : 2,
        "m" : 2,
        "n" : 2,
        "name" : "Matrix-2x2VW",
```

```
"rxArray" : [
	[
		{
			"aliasName" : "TopLeft-2x2",
			"deviceType" : "Receiver",
			"group" : [
				{
					"name" : "2x2-Wall",
					"sequence" : 2
				}
			],
			"online" : true,
			"rxstatus" : 1,
			"sequence" : 2,
			"trueName" : "IPEX2002-341B22801768",
			"txName" : "IPEX2001-341B2280175E"
		},
		{
			"aliasName" : "TopRight-2x2",
			"deviceType" : "Receiver",
			"group" : [
				{
					"name" : "2x2-Wall",
					"sequence" : 2
				}
			],
			"online" : true,
			"rxstatus" : 1,
			"sequence" : 1,
			"trueName" : "IPEX2002-341B22801763",
			"txName" : "IPEX2001-341B2280175E"
		}
	],
	[
		{
			"aliasName" : "BottomLeft-2x2",
			"deviceType" : "Receiver",
			"group" : [
				{
					"name" : "2x2-Wall",
					"sequence" : 2
				}
			],
			"online" : true,
			"rxstatus" : 1,
			"sequence" : 4,
			"trueName" : "IPEX2002-341B22801761",
			"txName" : "IPEX2001-341B2280175E"
		},
		{
			"aliasName" : "BottomRight-2x2",
			"deviceType" : "Receiver",
```

```
                        "group" : [
                                {
                                        "name" : "2x2-Wall",
                                        "sequence" : 2
                                }
                        ],
                        "online" : true,
                        "rxstatus" : 1,
                        "sequence" : 3,
                        "trueName" : "IPEX2002-341B22801769",
                        "txName" : "IPEX2001-341B2280175E"
                }
        ]
],
"sceneAutoApply" : false,
"sequence" : 1,
"txListArray" : [
        [
                {
                        "devices" : [
                                {
                                        "aliasName" : "iPad",
                                        "deviceType" : "Transmitter",
                                        "group" : [
                                                {
                                                        "name" : "ungrouped",
                                                        "sequence" : 1
                                                }
                                        ],
                                        "online" : true,
                                        "rxstatus" : 1,
                                        "sequence" : 1,
                                        "trueName" : "IPEX2001-341B22801758"
                                }
                        ]
                },
                {
                        "devices" : [
                                {
                                        "aliasName" : "iPad",
                                        "deviceType" : "Transmitter",
                                        "group" : [
                                                {
                                                        "name" : "ungrouped",
                                                        "sequence" : 1
                                                }
                                        ],
                                        "online" : true,
                                        "rxstatus" : 1,
                                        "sequence" : 1,
                                        "trueName" : "IPEX2001-341B22801758"
                                }
                        ]
                }
        ],
```

```
		[
			{
				"devices" : [
					{
						"aliasName" : "iPad",
						"deviceType" : "Transmitter",
						"group" : [
							{
								"name" : "ungrouped",
								"sequence" : 1
							}
						],
						"online" : true,
						"rxstatus" : 1,
						"sequence" : 1,
						"trueName" : "IPEX2001-341B22801758"
					}
				]
			},
			{
				"devices" : [
					{
						"aliasName" : "iPad",
						"deviceType" : "Transmitter",
						"group" : [
							{
								"name" : "ungrouped",
								"sequence" : 1
							}
						],
						"online" : true,
						"rxstatus" : 1,
						"sequence" : 1,
						"trueName" : "IPEX2001-341B22801758"
					}
				]
			}
		]
	],
	"vwConfigList" : [
		{
			"col_count" : 2,
			"mode" : "0",
			"name" : "2x2vw",
			"oh" : 0,
			"ow" : 0,
			"pos_col" : 0,
			"pos_row" : 0,
			"row_count" : 2,
			"vh" : 0,
			"vw" : 0
		}
	]
},
```

```
	{
		"group" : [
			{
				"name" : "Entrance",
				"sequence" : 3
			}
		],
		"layoutseq" : 1,
		"m" : 1,
		"n" : 1,
		"name" : "entrance-Entrance",
		"rxArray" : [
			[
				{
					"aliasName" : "Entrance",
					"deviceType" : "Receiver",
					"group" : [
						{
							"name" : "Entrance",
							"sequence" : 3
						}
					],
					"online" : true,
					"rxstatus" : 1,
					"sequence" : 1,
					"trueName" : "IPEX2002-341B22801765",
					"txName" : "IPEX2001-341B2280175E"
				}
			]
		],
		"sceneAutoApply" : false,
		"sequence" : 1,
		"txListArray" : [
			[
				{
					"devices" : [
						{
							"aliasName" : "iPad",
							"deviceType" : "Transmitter",
							"group" : [
								{
									"name" : "ungrouped",
									"sequence" : 1
								}
							],
							"online" : true,
							"rxstatus" : 1,
							"sequence" : 1,
							"trueName" : "IPEX2001-341B22801758"
						}
					]
				}
			]
		]
	}
]
```

device info

`config set device info` and `config get device info` send data in key-value format. *`key`* is the parameter and *`value`* is its value. The following table lists the parameters supported by devices and their value ranges. All parameters can be changed, unless otherwise stated.

Parameter	Description	Device
`name`	Device name. Read only. Format is "Device type-MAC address" such as IPEX2001-341B22FFFFB3.	All devices
`version`	Device software version. Read only. Format is v#.#.# such as `v2.5.6`.	All devices
`mac`	Device MAC address. Read only.	All devices
`ip_mode`	IP address obtain method. `autoip`: AutoIP `static`: Static IP `dhcp`: DHCP	All devices
`ip4addr`	IPv4 address. When ip_mode is `static`, IPv4 address can be changed.	All devices
`netmask`	IPv4 subnet mask. When ip_mode is `static`, IPv4 subnet mask can be changed.	All devices
`gateway`	IPv4 gateway. When ip_mode is `static`, IPv4 gateway can be changed.	All devices
`enc_rc_mode`	Data rate control method. `cbr` is CBR mode. `vbr` is VBR mode. `fixqp` is Fixed QP mode.	Encoder
`profile`	Encoding profile. bp is base profile. mp is main profile. hp is high profile.	Encoder
`cbr_avg_bitrate`	CBR encoding average rate. Unit is kbps. Data rate of IPEX2001 is less than or equal to 30720.	Encoder
`vbr_max_bitrate`	VBR encoding maximum rate. Unit is kbps. Data rate of IPEX2001 is less than or equal to 30720.	Encoder
`vbr_min_gp`	VBR minimum quantification parameters. Range is [0, 51].	Encoder
`vbr_max_gp`	VBR maximum quantification parameters. Range is [0, 51].	Encoder
`fixqp_iqp`	FixQP encoding mode I-frame quantification parameters. Range is [0, 51].	Encoder
`fixqp_pqp`	FixQP encoding mode P-frame quantification parameters. Range is [0, 51].	Encoder
`enc_gop`	GOP size. Range is [1, 65535]. There is one I-frame in a specific range.	Encoder
`enc_fps`	Frames per second. Range is [1, 60].	Encoder
`transport_type`	Streaming media encapsulation format. `raw` is private format. `ts` is MPEG-2 TS format.	Encoder
`audio.name`	Audio interface name. Read-only. Names like `linein1`, `linein2`, `lineout1` and `lineout2` are related to device hardware configuration.	All devices
`audio.mute`	Audio interface mute status. `true` is "mute". `false` is "unmute". For example, `audio.lineout1.mute=true`.	Decoder

matrix Commands

matrix set Commands Summary

Commands	Description
`matrix set`	Sets encoder to decoder matrix routes
`matrix video set`	Sets encoder to decoder video routes (5000 series only)
`matrix audio set`	Sets encoder to decoder audio routes (5000 series only)
`matrix usb set`	Sets encoder to decoder USB routes (5000 series only)

matrix set

`matrix set` sets the encoder to decoder matrix routes. Route groups are separated by commas. Every group starts with the encoder and ends with the decoder(s). Using `NULL` as the encoder will stop the video feed going to the decoder. When a decoder is in a video wall, this command is used to switch to another encoder but will not clear video wall settings.

`encoderx` and *`decoderx`* are device names, which may be the default name or an alias.

Command Structure

```
matrix set encoder1 decoder1 decoder2, encoder2 decoder3 decoder4,...
```

Response

```
matrix set encoder1 decoder1 decoder2, encoder2 decoder3 decoder4,...
```

Example

Route encoder IPEX2001-341B22FFFFC1 to decoders IPEX2002-341B22800316 and IPEX2002-341B22800309, encoder IPEX2001-341B22FFFFC2 to decoder IPEX2002-341B22800319, and encoder IPEX2001-341B22FFFFC3 to decoder IPEX2002-341B2280031A.

Command

```
matrix set IPEX2001-341B22FFFFC1 IPEX2002-341B22800316 IPEX2002-341B22800309, IPEX2001-341B22FFFFC2 IPEX2002-341B22800319, IPEX2001-341B22FFFFC3 IPEX2002-341B2280031A
```

Response

```
matrix set IPEX2001-341B22FFFFC1 IPEX2002-341B22800316 IPEX2002-341B22800309, IPEX2001-341B22FFFFC2 IPEX2002-341B22800319, IPEX2001-341B22FFFFC3 IPEX2002-341B2280031A
```

matrix video set (5000 series only)

`matrix video set` sets the encoder to decoder video routes for 5000 series products only. Route groups are separated by commas. Every group starts with the encoder and ends with the decoder(s). Using `NULL` as the encoder will stop the video feed going to the decoder.

`encoderx` and *`decoderx`* are device names, which may be the default name or an alias.

Command Structure

```
matrix video set encoder1 decoder1 decoder2, encoder2 decoder3 decoder4,...
```

Response

```
matrix video set encoder1 decoder1 decoder2, encoder2 decoder3 decoder4,...
```

Example

Route the video from encoder IPEX5001-341B22FFFFC to decoders IPEX5002-341B22800316, IPEX5002-341B22800309, IPEX5002-341B22800319 and IPEX5002-341B2280031A.

Command

```
matrix video set IPEX5001-341B22FFFFC2 IPEX5002-341B22800316 IPEX5002-341B22800309
IPEX5002-341B22800319 IPEX5002-341B2280031A
```

Response

```
matrix video set IPEX5001-341B22FFFFC2 IPEX5002-341B22800316 IPEX5002-341B22800309
IPEX5002-341B22800319 IPEX5002-341B2280031A
```

matrix audio set (5000 series only)

`matrix audio set` sets the encoder to decoder audio routes for 5000 series products only. Route groups are separated by commas. Every group starts with the encoder and ends with the decoder(s). Using `NULL` as the encoder will stop the audio feed going to the decoder.

`encoderx` and *`decoderx`* are device names, which may be the default name or an alias.

Command Structure

```
matrix audio set encoder1 decoder1 decoder2, encoder2 decoder3 decoder4,...
```

Response

```
matrix audio set encoder1 decoder1 decoder2, encoder2 decoder3 decoder4,...
```

Example

Route the audio from encoder IPEX5001-341B22FFFFC to decoders IPEX5002-341B22800316, IPEX5002-341B22800309, IPEX5002-341B22800319 and IPEX5002-341B2280031A.

Command

```
matrix audio set IPEX5001-341B22FFFFC2 IPEX5002-341B22800316 IPEX5002-341B22800309
IPEX5002-341B22800319 IPEX5002-341B2280031A
```

Response

```
matrix audio set IPEX5001-341B22FFFFC2 IPEX5002-341B22800316 IPEX5002-341B22800309
IPEX5002-341B22800319 IPEX5002-341B2280031A
```

matrix usb set (5000 series only)

`matrix usb set` sets the encoder to decoder USB routes for 5000 series products only. Route groups are separated by commas. Every group starts with the encoder and ends with the decoder(s). Using `NULL` as the encoder will break the USB connection coming from the decoder.

`encoderx` and *`decoderx`* are device names, which may be the default name or an alias.

Command Structure

```
matrix usb set encoder1 decoder1 decoder2, encoder2 decoder3 decoder4,...
```

Response

```
matrix usb set encoder1 decoder1 decoder2, encoder2 decoder3 decoder4,...
```

Example

Route the USB signal to encoder IPEX5001-341B22FFFFC from decoders IPEX5002-341B22800316, IPEX5002-341B22800309, IPEX5002-341B22800319 and IPEX5002-341B2280031A.

Command

```
matrix usb set IPEX5001-341B22FFFFC2 IPEX5002-341B22800316 IPEX5002-341B22800309
IPEX5002-341B22800319 IPEX5002-341B2280031A
```

Response

```
matrix usb set IPEX5001-341B22FFFFC2 IPEX5002-341B22800316 IPEX5002-341B22800309
IPEX5002-341B22800319 IPEX5002-341B2280031A
```

matrix get Commands Summary

Commands	Description
`matrix get`	Queries encoder to decoder matrix routes
`matrix video get`	Queries encoder to decoder video routes (5000 series only)
`matrix audio get`	Queries encoder to decoder audio routes (5000 series only)
`matrix usb get`	Queries encoder to decoder USB routes (5000 series only)

matrix get

`matrix get` queries the encoder to decoder matrix routes. Encoder to decoder routes are separated by lines based on the individual decoder. A `NULL` as the encoder indicates no video is going to the decoded. When a decoder is in a video wall, no video wall information is returned.

`encoderx` and *`decoderx`* are device names, which may be the default name or an alias.

Command Structure

```
matrix get
```

Response

```
matrix information:
encoder1 decoder1
encoder2 decoder3
encoder2 decoder4
...
```

Example

Query encoder to decoder matrix routes.

Command

```
matrix get
```

Response

```
matrix information:
IPEX5001-341B2243011A IPEX5002-341B22800BCD
IPEX5001-341B2243011A IPEX5002-341B22800BCE
IPEX5001-341B2243011A IPEX5002-341B22800BCA
null IPEX5002-341B22800BC6
```

matrix video get (5000 series only)

`matrix video get` queries the encoder to decoder video routes for 5000 series products only. Encoder to decoder routes are separated by lines based on the individual decoder. Defining specific decoders will respond with only the specified decoders. A `NULL` as the encoder indicates no video is going to the decoded. When a decoder is in a video wall, no video wall information is returned.

`encoderx` and *`decoderx`* are device names, which may be the default name or an alias.

Command Structure

```
matrix video get
```

Response

```
matrix video information:
encoder1 decoder1
encoder2 decoder3
encoder2 decoder4
...
```

Example 1

Query encoder to decoder video routes for decoders IPEX5002-341B22800BCD and IPEX5002-341B22800BCA.

Command

```
matrix video get IPEX5002-341B22800BCD IPEX5002-341B22800BCA
```

Response

```
matrix video information:
IPEX5001-341B2243011A IPEX5002-341B22800BCD
IPEX5001-341B2243011A IPEX5002-341B22800BCA
```

Example 2

Query encoder to decoder video routes for the system.

Command

```
matrix video get
```

Response

```
matrix video information:
IPEX5001-341B2243011A IPEX5002-341B22800BCD
IPEX5001-341B2243011A IPEX5002-341B22800BCE
IPEX5001-341B2243011A IPEX5002-341B22800BCA
null IPEX5002-341B22800BC6
```

matrix audio get (5000 series only)

`matrix audio get` queries the encoder to decoder audio routes for 5000 series products only. Encoder to decoder routes are separated by lines based on the individual decoder. Defining specific decoders will respond with only the specified decoders. A `NULL` as the encoder indicates no audio is going to the decoded. When a decoder is in a video wall, no video wall information is returned.

`encoderx` and *`decoderx`* are device names, which may be the default name or an alias.

Command Structure

```
matrix audio get
```

Response

```
matrix audio information:
encoder1 decoder1
encoder2 decoder3
encoder2 decoder4
...
```

Example 1

Query encoder to decoder audio routes for decoders IPEX5002-341B22800BCD and IPEX5002-341B22800BCA.

Command

```
matrix audio get IPEX5002-341B22800BCD IPEX5002-341B22800BCA
```

Response

```
matrix audio information:
IPEX5001-341B2243011A IPEX5002-341B22800BCD
IPEX5001-341B2243011A IPEX5002-341B22800BCA
```

Example 2

Query encoder to decoder audio routes for the system.

Command

```
matrix audio get
```

Response

```
matrix audio information:
IPEX5001-341B2243011A IPEX5002-341B22800BCD
IPEX5001-341B2243011A IPEX5002-341B22800BCE
IPEX5001-341B2243011A IPEX5002-341B22800BCA
null IPEX5002-341B22800BC6
```

matrix usb get (5000 series only)

`matrix usb get` queries the encoder to decoder USB routes for 5000 series products only. Encoder to decoder routes are separated by lines based on the individual decoder. Defining specific decoders will respond with only the specified decoders. A `NULL` as the encoder indicates no USB connection is made with the decoder.

`encoderx` and *`decoderx`* are device names, which may be the default name or an alias.

Command Structure

```
matrix usb get
```

Response

```
matrix usb information:
encoder1 decoder1
encoder2 decoder3
encoder2 decoder4
...
```

Example 1

Query encoder to decoder USB routes for decoders IPEX5002-341B22800BCD and IPEX5002-341B22800BCA.

Command

```
matrix usb get IPEX5002-341B22800BCD IPEX5002-341B22800BCA
```

Response

```
matrix usb information:
IPEX5001-341B2243011A IPEX5002-341B22800BCD
IPEX5001-341B2243011A IPEX5002-341B22800BCA
```

Example 2

Query encoder to decoder USB routes for the system.

Command

```
matrix video get
```

Response

```
matrix usb information:
IPEX5001-341B2243011A IPEX5002-341B22800BCD
IPEX5001-341B2243011A IPEX5002-341B22800BCE
IPEX5001-341B2243011A IPEX5002-341B22800BCA
null IPEX5002-341B22800BC6
```

vw Commands

vw Commands Summary

Commands	Description
`vw add`	Creates an n x m video wall configuration and assigns an encoder
`vw rm`	Removes a video wall configuration
`vw rm vwname decoderx`	Removes one or more decoders from a video wall configuration
`vw add position`	Adds one or more decoders to a video wall configuration
`vw add layout`	Creates an n x m video wall configuration and assigns an encoder and n x m decoders to it
`vw change decoderx encoderx`	Removes one decoder from a video wall configuration and sets an encoder to be the new source
`vw change vw-name encoderx`	Sets a new encoder as the video wall source
`vw bezelgap`	Sets bezel compensation parameters (5000 series only)
`vw get`	Queries all video wall configurations

vw add

`vw add` creates an n x m video wall configuration and assigns an encoder to the video wall. This command is used to create records in the IPEXCB but does not change the devices' working status.

`vw-name` is the video wall name and must be different from all others. *`n`* is the number of rows; *`m`* is the number of columns. *`encoderx`* is the encoder name, which may be the default name or an alias.

Command Structure

`vw add` *`vw-name n m encoderx`*

Response

`videowall item` *`vw-name`* `create and assign` *`encoderx`* `to it`

Example

Create a 2 x 2 video wall configuration *vwtest1* and assign encoder IPEX2001-341B2243011A.

Command

```
vw add vwtest1 2 2 IPEX2001-341B2243011A
```

Response

```
videowall item vwtest1 create and assign IPEX2001-341B2243011A to it
```

vw rm

`vw rm` removes a video wall configuration. This command is used to remove records in the IPEXCB but does not change the devices' working status. If the current video wall is removed, the active decoders will still play the current content.

`vw-name` is the video wall name.

Command Structure

`vw rm` *`vw-name`*

Response

`videowall item` *`vw-name`* `removed`

Example

Remove video wall configuration *vwtest1*.

Command

```
vw rm vwtest1
```

Response

```
videowall item vwtest1 removed
```

vw rm vwname decoderx

`vw rm vwname` *`decoderx`* removes one or more decoders from a video wall configuration. When the decoder is removed, it shows a full screen image of the current encoder.

`vw-name` is the video wall name. *`decoderx`* is the decoder name, which may be the default name or an alias.

Command Structure

`vw rm` *`vw-name decoderx1 decoderx2...`*

Response

`videowall config change:` *`decoderx1 decoderx2...`* `from` *`vw-name`*

Example

Remove decoders IPEX5002-341B22800BCE and IPEX5002-341B22800BCA from video wall *vwtest1*.

Command

```
vw rm vwtest1 IPEX5002-341B22800BCE IPEX5002-341B22800BCA
```

Response

```
videowall config change: remove IPEX5002-341B22800BCE IPEX5002-341B22800BCA from vwtest1
```

vw add position

`vw add position` adds one or more decoders to a video wall configuration. Once this command is executed, the decoder will start showing the current video wall content.

`vw-name` is the video wall name. *`decoderx`* is the decoder name, which may be the default name or an alias. *`x`* and *`y`* are the grid positions within the video wall; the top left position is 1 1.

Command Structure

```
vw add vw-name decoderx1 x1 y1 decoderx2 x2 y2...
```

Response

```
videowall item vw-name configuration added:
decoderx1 x1 y1
decoderx2 x2 y2
...
```

Example

Add four decoders to video wall configuration *vwtest2*.

Command

```
vw add vwtest2 IPEX5002-341B22800BCD 1 1 IPEX5002-341B22800BC6 1 2 IPEX5002-341B22800BCE
2 1 IPEX5002-341B22800BCA 2 2
```

Response

```
videowall item vwtest2 configuration added:
IPEX5002-341B22800BCD 1 1
IPEX5002-341B22800BC6 1 2
IPEX5002-341B22800BCE 2 1
IPEX5002-341B22800BCA 2 2
```

vw add layout

`vw add layout` creates an n x m video wall configuration and assigns an encoder and n x m decoders to it. Once this command is executed, the decoders will start showing the current video wall content.

`vw-name` is the video wall name. *`encoderx`* is the encoder name, which may be the default name or an alias. *`n`* is the number of rows; *`m`* is the number of columns. *`decoderx`* is the decoder name, which may be the default name or an alias. Decoders are automatically assigned positions in the video wall in the order their names are listed.

DEC11	DEC12	⋯	DEC1m
DEC21	DEC22	⋯	DEC2m
⋯	⋯	⋯	⋯
DECn1	DECn2	⋯	DECnm

Command Structure

`vw add` *`vw-name`* `layout` *`n m encoderx decoderx11 decoderx12 decoderx13 decoderx1m decoderx21 ... decoderxnm`*

Response

`videowall` *`vw-name`* `layout n*m` *`encoderx decoderx11 decoderx12 decoderx13 decoderx1m decoderx21... decoderxnm`*

Example

Create a 2 x 2 video wall configuration *vwtest3* which contains one encoder (IPEX5001-341B22430115) and four decoders (IPEX5002-341B22800BCD, IPEX5002-341B22800BC6, IPEX5002-341B22800BCE and IPEX5002-341B22800BCA).

Command

```
vw add vwtest3 layout 2 2 IPEX5001-341B22430115 IPEX5002-341B22800BCD IPEX5002-341B22800BC6 IPEX5002-341B22800BCE IPEX5002-341B22800BCA
```

Response

```
videowall vwtest3 layout 2*2 IPEX5001-341B22430115 IPEX5002-341B22800BCD IPEX5002-341B22800BC6 IPEX5002-341B22800BCE IPEX5002-341B22800BCA
```

vw change decoderx encoderx

`vw change` *`decoderx encoderx`* removes one decoder from a video wall configuration and sets an encoder to be the new source

`encoderx` and *`decoderx`* are device names, which may be the default name or an alias. If the encoder is set to `NULL`, no video will be displayed on the decoder.

Command Structure

`vw change` *`decoderx encoderx`*

Response

`videowall config clear:` *`decoderx`* `and connect to` *`encoderx`*

Example

Remove decoder IPEX5002-341B22800BCA from a video wall and switch this decoder to encoder IPEX5001-341B22430115.

Command

```
vw change IPEX5002-341B22800BCA IPEX5001-341B22430115
```

Response

```
videowall config clear: IPEX5002-341B22800BCA and connect to IPEX5001-341B22430115
```

vw change vw-name encoderx

`vw change` *`vw-name`* `encoderx` sets a new encoder as the video wall source.

`vw-name` is the name of the video wall. *`encoderx`* is the encoder name, which may be the default name or an alias. If the encoder is set to `NULL`, no video will be displayed on the video wall.

Command Structure

`vw change` *`vw-name encoderx`*

Response

`videowall` *`vw-name`* `tx connect to` *`encoderx`*

Example

Switch to encoder IPEX5001-341B22430115 for video wall *vwtest2*.

Command

```
vw change vwtest2 IPEX5001-341B22430115
```

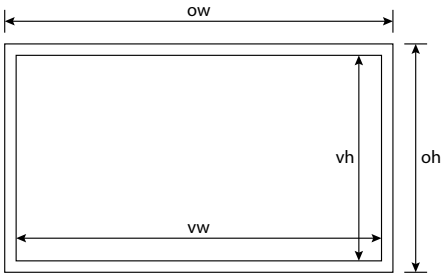
Response

```
videowall vwtest2 tx connect to IPEX5001-341B22430115
```

vw bezelgap (5000 series only)

`vw bezelgap` sets bezel compensation parameters for 5000 series decoders.

`vw-name` is video wall name. *`ow`* and *`oh`* are the overall width and height of display device including the bezel. *`vw`* and *`vh`* are the visible screen width and height. Values for ow, oh, vw and vh are in 0.1 mm increments.



Command Structure

```
vw bezelgap vw-name ow oh vw vh
```

Response

```
videowall vw-name's bezelgap: ow oh vw vh
```

Example

Set bezel compensation parameters with outside size 820 mm (ow) x 470 mm (oh) and screen size 800 mm (vw) x 450 mm (vh) in video wall *vwtest5*.

Command

```
vw bezelgap vwtest5 8200 4700 8000 4500
```

Response

```
videowall vwtest5's bezelgap: 8200 4700 8000 4500
```

vw get

`vw get` queries all video wall configurations.

vw-name1 and *vw-name2* are video wall names. *encoder1* is the encoder name of video wall *vw-name1*. *decoder1-11*, *decoder1-12*, *decoder1-21* and *decoder1-22* are decoders of video wall *vw-name1*. Numbers like "11" in *decoder1-11* and "12" in *decoder1-12* are the decoders' positions in the video wall but are not shown in the responses.

Command Structure

```
vw get
```

Response

```
Video wall information:
vw-name1 encoder1
Row 1: decoder1-11 decoder1-12
Row 2: decoder1-21 decoder1-22
...
vw-name2 encoder2
Row 1: decoder2-11 decoder2-12
Row 2: decoder2-21 decoder2-22
...
```

Example

Query all video wall configurations for the connected IPEXCB.

Command

```
vw get
```

Response

```
Video wall information:
2x2_vw IPEX2001-341B22801758
Row 1: IPEX2002-341B22801768 IPEX2002-341B22801763
Row 2: IPEX2002-341B22801761 IPEX2002-341B22801769
```

serial Command

`serial` sends commands to peripheral devices via serial ports on the encoders and decoders

`vw-name1` and *`vw-name2`* are video wall names. *`encoder1`* is the encoder name of video wall *`vw-name1`*. *`decoder1-11`*, *`decoder1-12`*, *`decoder1-21`* and *`decoder1-22`* are decoders of video wall *`vw-name1`*. Numbers like "11" in *`decoder1-11`* and "12" in *`decoder1-12`* are the decoders' positions in the video wall but are not shown in the responses.

Flags/Parameters	Description
`command-string`	Command to transmit to the peripheral devices excluding the double quotation marks.
`-b param`	Used to set serial working mode which contains baud rate, data bits, parity and stop bits. Default is *115200-8n1*. (Baud rate is 115200 bps, data bits are 8 bits,parity is "none", stop bits is "1".)
`-r {on|off}`	Used to set whether to add a carriage return to the end of this command, then send it to a peripheral device. By default, value is *on*.
`-h {on|off}`	Used to set whether to send commands in hexadecimal format. By default, value is *off*. Spaces must be used between hexadecimal bytes when the value is *on*.
`hostname1 hostname2`	Device names whose serial ports are used to send commands to peripheral devices connected to them. Multiple devices may be set at one time.

Command Structure

`serial [-b param] [-r {on|off}] [-h {on|off}] "command-string" hostname1 hostname2...`

Response

```
serial command received:
serial -b param –r {on|off} –h {on|off} "command-string" hostname1 hostname2...
```

Example 1

Have the serial port on IPEX2001-341B22FFCBC2 transmit characters *KA WE 4E CC* to a peripheral device in default mode (param is 115200-8n1 and command uses printable ASCII format) and add a carriage return in the end of this command.

Command

```
serial -b 115200-8n1 -r on "KA WE 4E CC" IPEX2001-341B22FFCBC2
```

Response

```
serial command received:
serial -b 115200-8n1 -r on "KA WE 4E CC" IPEX2001-341B22FFCBC2
```

Example 2

Have the serial ports on IPEX5002-341B22800BCD and IPEX5002-341B22800BCE send *AB 12 FD* in hexadecimal format to peripheral devices in default mode (param is 115200-8n1) and add a carriage return in the end of this command.

Command

```
serial -b 115200-8n1 -r on -h on "AB 12 FD" IPEX5002-341B22800BCD IPEX5002-341B22800BCE
```

Response

```
serial command received:
serial -b 115200-8n1 -r on -h on "AB 12 FD" IPEX5002-341B22800BCD IPEX5002-341B22800BCE
```

scene Commands

scene Commands Summary

Commands	Description
`scene get`	Queries all scene names
`scene active`	Enables a new scene in a video wall
`scene set`	Sets an encoder to a decoder in a scene of a video wall
`scene change scenename encoderx`	Sets one encoder to all decoders in a scene in a video wall
`scene connect scenename`	Set encoders to corresponding decoders of a scene in sequence

scene get

`scene get` queries all scene names.

Command Structure

```
scene get
```

Response

```
scene list:
scenename1 scenename2 scenename3...
```

Example

Query all scenes for the connected IPEXCB.

Command

```
scene get
```

Response

```
scene list:
Office-MeetingRoom Office-TrainingRoom Office-BreakRoom
```

scene active

`scene active` enables a new scene in a video wall. This action takes effect immediately.

Command Structure

`scene active` *`scenename`*

Response

`scene` *`scenename`* `active success`

Example

Enable a new scene, *Office-MeetingRoom*, on a video wall.

Command

```
scene active Office-MeetingRoom
```

Response

```
scene Office-MeetingRoom active success
```

scene set

`scene set` sets an encoder to a decoder in a scene of a video wall. This action makes the decoder display this source until `scene active` is executed.

`posX` and *`posY`* are the decoder coordinates within the video wall scene. *`encoderx`* is the encoder name, which may be the default name or an alias.

Command Structure

`scene set scenename` *`posX posY encoderx`*`...`

Response

`scene` *`scenename`*`'s source in [`*`posX`*`,`*`posY`*`] change to` *`encoderx`*

Example

Assign encoder *MediaPlayer1* to the decoder at position 1, 2 in scene *Office-MeetingRoom* of a video wall.

Command

```
scene set Office-MeetingRoom 1 2 MediaPlayer1
```

Response

```
Scene Office-MeetingRoom's source in [1 2] change to MediaPlayer1
```

scene change scenename encoderx

`scene change scenename encoderx` sets one encoder to all decoders in a scene in a video wall. This action makes the decoder display this source until `scene active` is executed.

Command Structure

`scene change` *`scenename encoderx`*

Response

`scene` *`scenename`*`'s tx change to` *`encoderx`*

Example

Assign a source (MediaPlayer1) to all decoders in *scene1* of a video wall.

Command

```
scene change scene1 MediaPlayer1
```

Response

```
scene scene1's tx change to MediaPlayer1
```

scene connect scenename

`scene connect scenename` encoders to corresponding decoders of a scene in the sequence they are listed. This command only functions once and will not be save to the IPEXCB.

Command Structure

`scene connect` *`scenename encoderx1 encoderx2`*`...`

Response

`scene connect` *`scenename encoderx1 encoderx2`*`... success`

Example

Assign sources (encoder1, encoder2, encoder3, encoder4) to the corresponding decoder of scene1 in sequence.

Command

```
scene connect scene1 encoder1 encoder2 encoder3 encoder4
```

Response

```
scene scene1's tx connect to encoder1 encoder2 encoder3 encoder4
```

notify Commands

`notify` commands are automatically sent to a third party controller. They show some status changes within the IP video system. A third party controller can capture this information from the session and offer it to the application layer. The commands in this section have no requests and responses.

notify endpoint

`notify endpoint` indicates that an encoder or decoder has come online or gone offline.

"+" indicates the device has come online. "–" indicates the device has gone offline.

Response

```
notify endpoint {+|-} hostname1 hostname2... {-|+} hostnameM hostnameN...
```

Example

IPEXCB informs a third party control device that IPEX2001-341B22800BB0 just got online.

Response

```
notify endpoint + IPEX2001-341B22800BB0
```

notify serialinfo

`notify serialinfo` indicates that data has been received on a device's RS232 port.

`hostname` is a device name which has received data. `hex` is hexadecimal format while `ascii` is ASCII format; they cannot be used in the same time. `\r` and *`\n`* are escape characters, meaning a carriage return anda line feed respectively.

`infolen` is the length of *`info`* in bytes. *`info`* is the actual data received. For ASCII data, `infolen` is the number of actual databytes received. For hexadecimal data, *(infolen+1)/3* is the number of actual data bytes received.

Response

```
notify serialinfo hostname {hex|ascii} infolen:\r\ninfo\r\n
```

Example 1

IPEX2002-341B228007CB's serial port receives 19 bytes which are hexadecimal characters *68 65 6C 11 6C 6F 11 22 33 44 00 55 66 77 99 AA CC DD FF*: (infolen is "56").

Response

```
notify serialinfo IPEX2002-341B228007CB hex 56:
68 65 6C 11 6C 6F 11 22 33 44 00 55 66 77 99 AA CC DD FF
```

Example 2

IPEX5002-341B22800BCA's serial port receives five characters "*12345*".

Response

```
notify serialinfo IPEX5002-341B22800BCA ascii 5:
12345
```

notify video (5000 series only)

`notify video` indicates a video feed has been lost or restored.

`VideoSourceName` is the encoder name that has lost or restored the video feed. This command only works with decoders.

Response

`notify video {lost|found} [(`*`VideoSourceName`*`)]`

Example

IPEX5001-341B22800BB0 has lost the video stream.

Response

```
notify video lost IPEX5001-341B22800BB0
```

IPLinx is a brand of:

11675 Ridgeline Drive
Colorado Springs, Colorado
80921 USA
Phone: 719-260-0061
Fax: 719-260-0075
Toll-Free: 800-530-8998
Email: supportlibav@libav.com