RS232 and Network Command Catalog

For JSON RPC/Pulse Based Projectors

For UDX

End User

Reference guide

1.7 2019-03-04



Table of contents

Pulse API	3
Introduction	3
Connecting to Pulse services	3
Quick start guide	4
Object and method naming	5
Type support	5
Parameters	6
Authentication	6
Service API	7
Methods	7
Properties	7
Signals	10
Notifications	12
Introspection API	13
Object changed signal	16
File endpoints	17
Programmers guide	19
Basic operation	19
Sources	21
Illumination	27
Picture settings	31
Warping with grid files	34
Blending with images	35
Black level adjustment with images	37
Environment information	38
ECO mode	41
Important note about the API documentation	41
Properties	42
Methods	169
Signals	233
Files	240
Property index	243
Method index	249
Signal index	252
File index	253

Pulse API

This document describes the application programmers interface to **Pulse** projectors.

How to connect to the projector, the communication protocol and a programmers guide is presented in the following section.

Introduction

The facade API is based on the JSON-RPC 2.0 protocol and provides access to **Pulse** services to clients. The services can be accessed through the network using the TCP/IP protocol, or using a RS232 serial cable.

Connecting to Pulse services

Network

If the projector is on a network, TCP/IP can be used to connect to **Pulse** services. The service is available on port number 9090.

Serial port

A serial cable can be connected to the projector in order to access the Pulse services.

Connect the projector and host using a standard serial cable with 9-pin female to the host, and 9-pin male to the projector. Pin 2 connects to pin 2, pin 3 connects to pin 3 and pin 5 connects to pin 5.

RS232 Communication Parameters

Parameter	Value
Baud rate	19200
Parity	None
Data bits	8
Stop bits	1
Flow control	None

Quick start guide

The following sections are examples of frequently used commands, just to get you started. This assumes that the connection is set up as described in the previous chapter. The type of connection is not important. The same commands are available for all connection types.

Power on projector

Request

```
{
  "jsonrpc": "2.0",
  "method": "system.poweron"
}
```

Power off projector

Request

```
{
  "jsonrpc": "2.0",
  "method": "system.poweroff"
}
```

Select DisplayPort 1 as input source

Request

```
{
  "jsonrpc": "2.0",
  "method": "property.set",
  "params": {
     "property": "image.window.main.source",
     "value": "DisplayPort 1"
  }
}
```

Select **HDMI** as input source

```
{
  "jsonrpc": "2.0",
  "method": "property.set",
  "params": {
     "property": "image.window.main.source",
     "value": "HDMI"
  }
}
```

Object and method naming

Objects and members are named using dot notation in lowercase format (JavaScript-like notation). Members are either a method, property, signal or object. A typical method part of an invocation will then look like:

```
method: "foo.echo"
```

If there are more than one object of a "kind", it may be modeled and notated like:

```
tempctrl.fans
tempctrl.fans.mainfan
tempctrl.fans.lampblower
```

In the example above it is possible to get all fans by introspecting the object represented by temptrl.fans. Example: accessing the rpm property of the mainfan:

```
tempctrl.fans.mainfan.rpm
```

Type support

- · Basic types
 - string (e.g. "hello")
 - integer (e.g 114)
 - float (e.g 3.141592653589793)
 - boolean (e.g true)
- · Container types
 - array (e.g ["hello", "world"])
 - object (e.g {"name": "Johnny", "age": 30, "children": ["Agnes", "Tim"] })
 - dictionary with string key (e.g gold medals in 2018 Winter Olympics Peyongchang {"Norway": 13, "Germany": 13 })

Parameters

All parameters are passed by name, but the position or order of the parameters doesn't matter.

```
Thus:
  "jsonrpc": "2.0",
  "method": "property.set",
  "params": {
    "property": "image.window.main.source",
    "value": "DisplayPort 1"
  }
}
is exactly the same as:
{
  "method": "property.set",
  "jsonrpc": "2.0",
  "params": {
    "value": "DisplayPort 1",
    "property": "image.window.main.source"
}
```

Authentication

A client session must start with an authentication request containing a secret pass code. The purpose of the authentication protocol is to set the user access level. Authentication is only necessary when a higher level than normal end user is required. For normal end user access the authentication can be skipped. To authenticate with the server use the following type of request.

Request Response

```
{
    "jsonrpc": "2.0",
    "method": "authenticate",
    "params": {
        "code": 98765
    },
    "id": 1
}
```

Service API

Property	Туре	Required	Comments
jsonrpc	string	yes	2.0
method	string	yes	see below
params			see below
id	string number	no	Request identifier
error	object	yes, if error	Error object - see JSON-RPC 2.0

Methods

Method invocation API

Request

Response

```
{
    "jsonrpc": "2.0",
    "method": "ledctrl.blink",
    "params": {
        "led": "systemstatus",
        "color": "red",
        "period": 42
},
    "id": 3
}
```

Properties

API for setting and getting property values

Set value of a property

Response

```
{
    "jsonrpc": "2.0",
    "method": "property.set",
    "params": {
        "property": "objectname.propertyname",
        "value": 100
},
    "id": 3
}
```

It is best practice to wait for the confirmation of the property set before setting the same property again. Continuously setting the same property without waiting for confirmation may flood the server with unnecessary request and may reduce performance.

Read the value of a property

Request

Response

```
{
    "jsonrpc": "2.0",
    "method": "property.get",
    "params": {
        "property": "objectname.propertyname"
    }
},
    "id": 4
}
```

Read values of multiple properties

Request

Response

```
"jsonrpc": "2.0",
                                  "jsonrpc": "2.0",
  "method": "property.get",
                                  "result": {
  "params": {
                                    "image.brightness": 0,
                                    "image.contrast": 1
    "property": [
      "image.brightness",
                                  },
                                  "id": 5
      "image.contrast"
    1
 },
  "id": 5
}
```

Observe changes on one property

For change notifications, see **Notifications**

Request

Response

```
{
    "jsonrpc": "2.0",
    "method": "property.subscribe",
    "params": {
        "property": "image.brightness"
    },
    "id": 6
}
```

Observe changes on multiple properties

For change notifications, see Notifications

Request

Response

```
{
    "jsonrpc": "2.0",
    "method": "property.subscribe",
    "params": {
        "property": [
            "image.brightness",
            "image.contrast"
    ]
},
    "id": 7
}
```

Stop observing one property

For change notifications, see **Notifications**

Response

```
{
   "jsonrpc": "2.0",
   "method": "property.unsubscribe",
   "params": {
        "property": "image.brightness"
   },
   "id": 8
}
```

Stop observing multiple properties

For change notifications, see **Notifications**

Request

Response

Signals

Subscribe to a signal

For change notifications, see Notifications

Response

```
{
    "jsonrpc": "2.0",
    "method": "signal.subscribe",
    "params": {
        "signal": "modelupdated"
    }
},
    "id": 10
}
```

Subscribe to multiple signals

For change notifications, see Notifications

Request

Response

Unsubscribe from a signal

For change notifications, see **Notifications**

Request

Response

```
{
    "jsonrpc": "2.0",
    "method": "signal.unsubscribe",
    "params": {
        "signal": "modelupdated"
    }
},
    "id": 12
}
```

Unsubscribe from multiple signals

For change notifications, see Notifications

Request

Response

Notifications

The client has to implement the notification API to retrieve notifications. Notification messages will not have an **id** and no response message must be returned.

Properties

The client must implement the property.changed function which receives an array of property/value pairs.

Notification

Signals

The client must implement the signal.callback function which receives an array of signal/argument-list pairs.

Notification

Introspection API

Read metadata method

Metadata of available objects (methods, properties, signals) can be read out. The data is restricted by the client's authenticated access level. A typical usage for metadata is to set up OSD menus, etc. The format of the metadata is described here: API introspection data format.

Introspection API (recursive)

Response

```
"jsonrpc": "2.0",
                               "jsonrpc": "2.0",
  "method": "introspect",
                               "result": {
  "params": {
                                 "object": "foo",
    "object": "foo",
                                 "methods": [
    "recursive": true,
    "id": 1
                                     "name": "echo"
  }
                                   }
}
                                 ],
                                 "more": ".... see API introspection data
                               },
                               "id": 1
- or -
```

```
{
    "jsonrpc": "2.0",
    "method": "introspect",
    "params": [
        "foo",
        true
    ],
    "id": 1
}
```

Introspection API (non recursive)

Response

```
"jsonrpc": "2.0",
                               "jsonrpc": "2.0",
  "method": "introspect",
                               "result": {
  "params": {
                                 "name": "motors",
    "object": "motors",
                                 "objects": [
    "recursive": false,
    "id": 2
                                     "name": "motors.motor1"
  }
                                   },
}
                                     "name": "motors.motor2"
                                   },
                                     "name": "motors.motor3"
                                 ]
                               },
                               "id": 2
                             }
```

```
-or-

{
    "jsonrpc": "2.0",
    "method": "introspect",
    "params": [
        "motors",
        false
],
    "id": 2
```

Property	Туре	Required	Comments
jsonrpc	string	yes	2.0
method	string	yes	introspect
params	<pre>object {"object": string}</pre>	no (default =	"object": name of object to introspect (dot notation allowed), default/empty will introspect everything. The object and string notations are equivalents.

Property	Туре	Required	Comments
	{"recursive": bool}	no (default=true)	recursive": if false then only object names are listed (one level). This is convenient if you want to list collections of objects
id	string number	no	Request identifier
result	object	yes	The full format of the result is described here API Introspection data format
error	object	if error	Error object - see JSON-RPC 2.0

Object changed signal

The introspect API provides a signal that triggers when new objects arrive, or when objects are removed. The name of the signal is: modelupdated. (See section Signals on how to subscribe and unsubscribe to signals.)

Subscribe to the model updated signal

Request Response

```
{
    "jsonrpc": "2.0",
    "method": "signal.subscribe",
    "params": {
        "signal": "modelupdated"
    }
},
    "id": 2
}
```

Callback method on client side

Notification

Argument	Type	Decription
object	string	Name of object
isnew	bool	true object is new false object is lost

File endpoints

Some objects provide endpoints for uploading or downloading various data types. For example, a warp grid can be uploaded to the warp engine. The objects that provide end points are found in the documentation below under the File endpoints headings.

To download a file from the projector, you must enter its URL. The URL is constructed from the following parts:

- http://
- Address of the projector. E.g. 192.168.1.100
- /api
- File endpoint. E.g: /image/processing/warp/file/transfer

This will give you a URL that looks like this: http://192.168.100/api/image/processing/warp/file/transfer

Entering this URL in a browser will trigger a download from the projector and save the current warp grid to your download folder.

You can also use the curl program to do the same. E.g: curl -O -J http://api/image/processing/warp/file/transfer

Note that not all endpoints supports downloading the current file. In those cases you need to specify which file to download. E.g: http://192.168.1.100/api/image/processing/warp/file/transfer/warpgrid.xml

To upload a file to the projector, use the **cur1** program, or some other tool that supports HTTP upload. To upload a warp grid to the projector from your local drive, enter the following command:

curl -F file=@warpgrid.xml http://192.168.1.100/api/image/processing/warp/file/transfer

You can also specify -X POST to the command, but in this case that's implied.

Programmers guide

This section describes common tasks for controlling the projector. For example, selecting input source and adjusting image properties.

Basic operation

This chapter describes basic operation such as powering on/off the projector.

Projector state

To get the current operation state from the projector, use the following command.

Request

Response

```
{
    "jsonrpc": "2.0",
    "method": "property.get",
    "params": {
        "property": "system.state"
    }
},
    "id": 1
}
```

The resulting value will be one of the following:

- "boot" the projector is booting up
- "eco" the projector is in ECO/power save mode
- "standby" the projector is in standby mode
- "ready" the projector is in ready mode
- "conditioning" the projector is warming up
- "on" the projector is on
- "deconditioning" the projector is cooling down

To be notified when the state changes, a subscription must be requested, as shown in the following example.

Response

```
{
  "jsonrpc": "2.0",
  "method": "property.subscribe",
  "params": {
     "property": "system.state"
  }
},
  "id": 2
}
```

When ever there is a change in the state, the server will notify the client as shown in the next example.

Notification

Power on

To power on the projector, issue the following request.

Request

Response

```
{
    "jsonrpc": "2.0",
    "method": "system.poweron",
    "params": {
        "property": "system.state"
    }
},
    "id": 3
}
```

Notice that the result is null. This is not an error, it's just that the method does not return a proper result. If there was an error, the response would contain an error member that contains information about the error that

occurred. Also notice that this method does not need any arguments, so the params member does not need to be present. Nothing bad happens if the params member is present. It will just be ignored.

If the projector already is on, or if it's in transition between states, nothing will happen. Therefore, it's good practice to verify that the projector state is either **standby** or **ready** before issuing the power on command.

Power off

To power on the projector, issue the following request.

Request { "jsonrpc": "2.0", "method": "system.poweroff", "params": { "property": "system.state" } }, "id": 4 }

If the projector already is off, or if it's in transition between states, nothing will happen. Therefore, it's good practice to verify that the projector state is **on** before issuing the power off command.

Sources

The source input management is made up windows, sources and connectors. Each window has a source attached, and each source is made up of one or more connectors.

Active source

To get the name of the currently active source, use the following request.

Request Response

```
{
    "jsonrpc": "2.0",
    "method": "property.get",
    "params": {
        "property": "image.window.main.source"
    }
},
    "id": 0
}
```

List available sources

To get a list of available sources, use the following request.

Request Response

```
{
                                     {
                                        "jsonrpc": "2.0",
  "jsonrpc": "2.0",
  "method": "image.source.list",
                                       "id": 1,
  "id": 1
                                       "result": [
}
                                         "DVI 1",
                                          "DVI 2",
                                          "DisplayPort 1",
                                         "DisplayPort 2",
                                         "Dual DVI",
                                         "Dual DisplayPort",
                                         "Dual Head DVI",
                                          "Dual Head DisplayPort",
                                          "HDBaseT",
                                          "HDMI",
                                          "SDI"
                                     }
```

The response contains a list of all the available source names. The list contents will vary depending on the projector model.

Set the active source

To set the active source, first get the list of available sources as shown in the previous section. Then use the following request.

Request

Response

```
{
    "jsonrpc": "2.0",
    "method": "property.2et",
    "params": {
        "property": "image.window.main.source",
        "value": "DisplayPort 1"
    },
    "id": 2
}
```

Connectors

The connectors are the physical input connectors of the projector. Available connectors depend on the projector

model.

To list all the available connectors, use the following request.

Request

Response

```
{
  "jsonrpc": "2.0",
                                           "jsonrpc": "2.0",
  "method": "image.connector.list",
                                           "id": 3,
  "id": 3
                                           "result": [
                                              "DVI 1",
}
                                             "DVI 2",
                                              "DisplayPort 1",
                                              "DisplayPort 2",
                                              "HDBaseT",
                                              "HDMI",
                                              "SDI"
                                           ]
                                         }
```

To list the connectors used by a specific source, the source **object** name must be acquired. This can be done by translating the source name, or by introspection.

The easiest way is to translate the source name. Given the source name of <code>DisplayPort 1</code>, remove all non word characters and convert to all lower case characters. This can be done quite easy by using regular expressions. Example in <code>JavaScript</code> shown below.

```
const sourceName = 'DisplayPort 1';
const objectName = sourceName.replace(/\W/g, '').toLowerCase();
/* objectName is now 'displayport1' */
```

Now that we have the source object name, we can call the method that lists all the connectors used by this source.

Request Response

The result of the method is an array of connector information. This information contains the connector name and the grid position of the connector, which is useful when multiple connectors are used.

Source signal

Given a connector name of <code>DisplayPort 1</code>, this translated to a connector name of <code>displayport1</code>. We can then get signal information by using the following request.

Request Response

```
"jsonrpc": "2.0",
                                                                  "jsonrpc": "2.0",
                                                                  "id": 5,
"method": "property.get",
"params": \{
                                                                  "result": {
  "property":
                                                                    "active": true,
"image.connector.displayport1.detectedsignal"
                                                                    "name": "2560x1600 @ 50.10Hz",
                                                                    "vertical_total": 1638,
},
"id": 5
                                                                    "horizontal_total": 2720,
                                                                    "vertical_resolution": 1600,
                                                                    "horizontal_resolution": 2560,
                                                                    "vertical_sync_width": 6,
                                                                    "vertical_front_porch": 3,
                                                                    "vertical_back_porch": 29,
                                                                    "horizontal_sync_width": 32,
                                                                    "horizontal_front_porch": 48,
                                                                    "horizontal_back_porch": 80,
                                                                    "horizontal_frequency":
                                                                 82068.11653672549,
                                                                    "vertical_frequency":
                                                                 50.102710556641114,
                                                                    "pixel_rate": 223222961,
                                                                    "scan": "Progressive",
                                                                    "bits_per_component": 10,
                                                                    "color_space": "RGB",
                                                                    "signal_range": "0-255",
                                                                    "chroma_sampling": "4:4:4",
                                                                    "gamma_type": "POWER"
                                                                  }
```

Source and signal updates

To get updates when a new source is selected, or the signal on a connector change, you must listen forchanges on a number of properties.

Subscribe to the source property of the window

Response

```
{
    "jsonrpc": "2.0",
    "method": "property.subscribe",
    "params": {
        "property": "image.window.main.source"
    }
},
    "id": 6
}
```

This will generate a property change notification when ever there is a change in the active source. E.g when the user switches from DisplayPort 1 to DisplayPort 2.

The JSON RPC client must therefore implement the property change listener in order to process the notification.

The JSON RPC servier will send notifications as shown below.

Notification

```
{
  "jsonrpc": "2.0",
  "method": "property.changed",
  "params": {
     "property": [
          {
                "image.window.main.source": ""
          }
        ]
    }
}
```

Notification

Note that two notifications are delivered in this case. First, when the previously selected source is deselected. Next, when the new source is selected.

Connector signal updates

To get updates when there is a change in the signal on sources connectors, each connector of the sourcemust have a listener for the signal detection property of the connector.

The recommended way to do this is to reflect the source and connector structure of the server in the client application.

That means:

- Store all the source names and object names
 - Callimage.source.list
 - Translate the source names into source object names as shown in the beginning of the chapter
- For each source object
 - Call image.source.[name].listconnectors
 - Translate the connector names to connector object names
 - Maintain a list of connectors per source
- For each connector object
 - Subscribe to image.connector.[name].detectedsignal

When the notifications are received, match up the connector names with the connectors used by the active source and show the new information to the user.

Keep in mind that notifications are only sent when there is an actual change in a value. Simply subscribing to a property does not get the current value. To get the current value use the property.get method and specify which property you are interested in.

Illumination

This section describes how to set the lamp/LED/laser power of the projector.

Illumination state

To read the state of the illumination use the following commands.

Response

```
{
    "jsonrpc": "2.0",
    "method": "property.get",
    "params": {
        "property": "illumination.state"
    },
    "id": 0
}
```

The result will be either **On** or **Off**. To receive notifications when the state changes, you need to subscribe. The following commands will request change notifications for the illumination state.

Request

Response

```
{
    "jsonrpc": "2.0",
    "method": "property.subscribe",
    "params": {
        "property": "illumination.state"
    }
},
    "id": 1
}
```

The client needs to implement the property change listener to receive notifications.

When a change happens, the server will send the following notifications to the client.

Notification

```
{
  "jsonrpc": "2.0",
  "method": "property.changed",
  "params": {
      "property": [
            {
                  "illumination.state": "On"
            }
        ]
    }
}
```

Illumination sources

Different projectors will have different types of illumination sources. Some will have lasers, others will have LEDs, or a combination of both, and yet others may have xenon or UHP lamps.

To query the available sources, you must do an **introspection** of the **illumination** object. The following command shows an example of that.

Request Response

```
{
    "jsonrpc": "2.0",
    "method": "introspect",
    "params": {
        "property": "illumination.sources",
        "recursive": false
    },
    "id": 2
}

// "id": 2

// "name": "illuminations.sources.laser"
    }

// "id": 2

// "name": "illuminations.sources.laser"
// "id": 2

// "id": 2

// "name": "illuminations.sources.laser"
// "id": 2

// "id": 2

// "name": "illuminations.sources.laser"
// "id": 2

// "result": {

// "objects": [

// "name": "illuminations.sources.laser"
// "id": 2

// "result": {

// "objects": [

// "name": "illuminations.sources.laser"
// "id": 2

// "name": "illuminations.sources.laser"
// "id": 2

// "id":
```

From the response, we can see that this projector has a laser illumination source. Each source has properties that give you information about the minimum, maximum and the current illumination power level.

The current power level can be read or written, the minimum and maxium power level are read only, but they are dynamic values and may change depending on the setting of the projector. The lens type and lens position may also affect the power levels.

To read the current power level, use the following commands.

Request Response

```
{
    "jsonrpc": "2.0",
    "method": "property.get",
    "params": {
        "property": "illumination.sources.laser.power"
    }
},
    "id": 3
}
```

To be notified of changes in the power level, you must subscribe to property changes.

Response

```
{
    "jsonrpc": "2.0",
    "method": "property.subscribe",
    "params": {
        "property": [
            "illumination.sources.laser.power"
        ]
    },
    "id": 4
}
```

To set the value of the laser power, use the following command.

Request Response

```
{
    "jsonrpc": "2.0",
    "method": "property.set",
    "params": {
        "property": "illumination.sources.laser.power",
        "value": 40
},
    "id": 5
}
```

After the confirmation of the value, a change notification is also sent.

Notification

To get the minimum and maximum power levels, use the following commands.

Request Response

```
"jsonrpc": "2.0",
                                                             "jsonrpc": "2.0",
                                                             "id": 5,
  "method": "property.get",
  "params": {
                                                             "result": 100
    "property": "illumination.sources.laser.power"
 },
  "id": 5
}
  "jsonrpc": "2.0",
                                                             "jsonrpc": "2.0",
  "method": "property.get",
                                                             "id": 6,
                                                             "result": 0
  "params": {
    "property": "illumination.sources.laser.minpower"
 },
  "id": 6
```

Picture settings

The image service has properties for controlling the image appearance. Like brightness, contrast, saturation and gamma.

This section describes how to set basic picture settings.

Brightness

Before setting any values it is a good idea to get some information about the properties. This is done with introspection.

Individual properties can not be introspected, so we must ask the owner of the property. In this case it is the image service.

To perform introspection of the image service, issue the following request.

Response

```
"jsonrpc": "2.0",
                            "jsonrpc": "2.0",
  "method":
                            "id": 6,
 "introspect",
                            "result": {
  "params": {
                               "name": "image",
    "object": "image",
                               "properties": [
    "recursive": false
 },
                                   "name": "brightness",
  "id": 6
                                   "type": {
                                     "base": "float",
}
                                     "min": -1,
                                     "max": 1,
                                     "step-size": 1,
                                     "precision": 0.01
                                  },
                                   "access": "READ_WRITE",
                                   "description": "Image brightness/offset. The value is normalized, 0 is
                          default, 1 is 100% offset."
                                },
                                {
                                  "more": "...trimmed for brevity"
                               "objects": [
                                   "more": "...trimmed for brevity"
                          }
```

The result contains information about all the properties, objects and signals belonging to the image service.

Here we are only interested in the brightness property, so the rest of the response has been trimmed.

We can see that the **brightness** property has a **type** which is **float**. It also has some constraints that define the minimum and maximum values for the property. This information can be used to prevent sending out of range values to the server, and also give hints to the user when e.g using a slider to control the brightness value.

The type also has a step-size. This is a hint that can be used when incrementing or decrementing the value in a GUI application. The step-size is used as a factor to the precision. E.g for brightness we have a precision of 0.01 and a step-size of 1. That means that when ever the user wants to increment the value, we can add (step-size x precision) to the current value.

To get the current brightness value, use the following request.

Request

Response

```
{
   "jsonrpc": "2.0",
   "method": "property.get",
   "params": {
        "property": "image.brightness"
   }
},
   "id": 7
}
```

To receive notification when the brightness value change, use the following subscription.

Request

Response

```
{
    "jsonrpc": "2.0",
    "method": "property.subscribe",
    "params": {
        "property": [
            "image.brightness"
        ]
    },
    "id": 8
}
```

The set the brightness value, use the following request.

Request

Response

```
{
    "jsonrpc": "2.0",
    "method": "property.set",
    "params": {
        "property": "image.brightness",
        "value": 0.15
},
    "id": 9
}
```

In addition to the confirmation from the property.set method, a notification is also sent.

Notification

The set the other picture settings follow the same procedure as for brightness.

Warping with grid files

To globally enable warp, use the following command.

Request Response

```
{
   "jsonrpc": "2.0",
   "method": "property.set",
   "params": {
        "property": "image.processing.warp.enable",
        "value": true
},
   "id": 10
}
```

Upload warp file

Send the warp grid file through HTTP POST. The following example is using curl to upload a file to the projector at address 192.168.1.100.



curl -X POST -F file=@warp.xml http://192.168.1.100/api/image/processing/warp/file/transfer

Note that -X POST can be omitted since this is implied when using -F.

To select/activate the uploaded file, use the following command.

Request Response

```
{
    "jsonrpc": "2.0",
    "method": "property.set",
    "params": {
        "property": "image.processing.warp.file.selected",
        "value": "warp.xml"
},
    "id": 11
}
```

Finally, the grid file warping must be enabled. To do this, use the following command.

Request Response

```
{
   "jsonrpc": "2.0",
   "method": "property.set",
   "params": {
        "property": "image.processing.warp.file.enable",
        "value": true
},
   "id": 12
}
```

Warp file format

The warp file format is the same as on the MCM500/400.

Blending with images

In a multi channel setup, blending is required in order to get a seamless transition between the channels.

For this, blend edges can be configured, or one can use pre generated blend masks that are uploaded to the projector.

Uploading a blend mask

Blend masks are grayscale images in either 8 bit or 16 bit pixel resolution. The size of the blend mask must match the resolution of the blend layer of the projector. The following table shows the possible sizes.

Projector resolution	Mask resolution
WUXGA	1920 x 1200

Projector resolution	Mask resolution
WQXGA	1280 x 800
4K	1280 × 800
4K Cinemascope	1280 x 540

To upload a blend mask, use the following shell command.



curl -X POST -F file=@mask.png http://192.168.1.100/api/image/processing/blend/file/transfer

In the example above we have a blend mask in the PNG format called mask.png and the projector has the IP address of 192.168.1.100. To select the blend file that was just uploaded, us the following request.

Request Response

```
{
    "jsonrpc": "2.0",
    "method": "property.set",
    "params": {
        "property": "image.processing.blend.file.selected",
        "value": "mask.png"
},
    "id": 13
}
```

To enable the blend mask use the following request.

Request Response

```
{
    "jsonrpc": "2.0",
    "method": "property.set",
    "params": {
        "property": "image.processing.blend.file.enable",
        "value": true
},
    "id": 14
}
```

Supported image formats

The supported image formats are:

- PNG up to 16 bit
- JPEG
- TIFF

The interface only supports gray scale images, but will accept color images. It will then only use the blue channel. This is to support gray scale images saved as RGB images, we don't support the using the full colour information.

Black level adjustment with images

In a multi channel setup, blending is required in order to get a seamless transition between the channels. To compensate for the extra light in the blended regions, the black level can be increased in the non-blended regions to get a uniform black level across all channels. Black levels can be configured by using the basic controls to specify linear edges, or one can use pre generated images that are uploaded to the projector.

Uploading a black level mask

Black level masks are grayscale images in either 8 bit or 16 bit pixel resolution. The size of the black level mask must match the resolution of the black level layer of the projector. The following table shows the possible sizes.

Projector resolution	Mask resolution
WUXGA	1920 x 1200
WQXGA	1280 x 800
4K	1280 x 800
4K Cinemascope	1280 x 540

To upload a black level mask, use the following shell command.



curl -X POST -F file=@blacklevel.png http://192.168.1.100/api/image/processing/blacklevel/file/transfer

In the example above we have a black level mask in the PNG format called blacklevel.png and the projector has the IP address of 192.168.1.100.

To select the black level file that was just uploaded, use the following request.

Request Response

```
{
  "jsonrpc": "2.0",
  "method": "property.set",
  "params": {
    "property": "image.processing.blacklevel.file.selected",
    "value": "blacklevel.png"
},
  "id": 15
}
```

To enable the black level mask use the following request.

Request Response

```
{
    "jsonrpc": "2.0",
    "method": "property.set",
    "params": {
        "property": "image.processing.blacklevel.file.enable",
        "value": true
},
    "id": 16
}
```

Supported image formats

The supported image formats are:

- PNG up to 16 bit
- JPEG
- TIFF

The interface only supports gray scale, but will accept colour images. It will then only use the blue channel. This is to support gray scale images saved as RGB images. We don't support using the full color information.

Environment information

The environment service manages a lot of information in order to keep running the projector at the optimal operating conditions. Among the information available are fan speeds, temperatures and voltages. This section describes an easy method to get information from the environment service.

Temperatures

Response

To get a snapshot or current reading of all the avilable temperature sensors, use the following request.

Request

{ "jsonrpc": "2.0", "jsonrpc": "2.0", "method": "environment.getcontrolblocks", "id": 18, "params": { "result": { "type": "Sensor", "environment.laser.board0.bank0.temperature": 35.3, "valuetype": "Temperature" "environment.laser.board0.bank1.temperature": 34.8, }, "environment.laser.board0.bank2.temperature": 35.3, "id": 18 "environment.laser.board0.heatsink0.temperature": 35, } "environment.laser.board0.heatsink1.temperature": 37.6, "environment.laser.board0.heatsink2.temperature": 40.4, "environment.laser.board1.bank0.temperature": 36.6, "environment.laser.board1.bank1.temperature": 36.2, "environment.laser.board1.bank2.temperature": 36.4, "environment.laser.board1.heatsink0.temperature": 34.7, "environment.laser.board1.heatsink1.temperature": 34.9, "environment.laser.board1.heatsink2.temperature": 36.5, "environment.temperature.cyclon5": 47.6, "environment.temperature.imx6": 40.1, "environment.temperature.inlet": 25.5, "environment.temperature.mainboard": 40.4, "environment.temperature.mainpower": 37.6,

The result is in the form of a dictionary, where the key is the name of the sensor and the value is the temperature reading.

}

"environment.temperature.outlet": 29.4,
"environment.temperature.scalerfpga": 52.8

Fan speeds

To get fan speeds, use the following request.

Request

Response

```
"jsonrpc": "2.0",
                                                  "jsonrpc": "2.0",
  "method": "environment.getcontrolblocks",
                                                  "id": 19,
                                                  "result": {
  "params": {
                                                    "environment.fan.ar1.tacho": 1800,
    "type": "Sensor",
    "valuetype": "Speed"
                                                    "environment.fan.ar2.tacho": 1850,
 },
                                                    "environment.fan.ar3.tacho": 1750,
  "id": 19
                                                    "environment.fan.ar4.tacho": 1800,
}
                                                    "environment.fan.ar5.tacho": 1800,
                                                    "environment.fan.driver.tacho": 2300,
                                                    "environment.fan.optics.tacho": 2600,
                                                    "environment.fan.pcb.tacho": 1400,
                                                    "environment.fan.phosphorleft.tacho": 3850,
                                                    "environment.fan.phosphorright.tacho": 3800,
                                                    "environment.fan.psu.tacho": 1450
                                               }
```

The result is in the form of a dictionary, where the key is the name of the sensor, a fan in this case, and the value is the fan speed reading.

Other environment info

Other environment info is also available and follows the same patterns as for temperatures and fan speeds. Use the **environment.getcontrolblocks** method with different sensor types and different sensor value types to get the desired data.

The applicable sensor types are:

- Sensor
- Filter
- Controller
- Actuator
- Alarm
- GenericBlock

The applicable sensor value types are:

- Temperature
- Speed
- PWM
- Voltage
- Current
- Power
- Altitude
- Pressure
- Humidity

- ADC
- Coordinate
- Peltier
- Waveform
- Average
- Delay
- Difference
- Interpolation
- Limit

- Median
- Noise
- Weighting
- Comparison
- Threshold
- Formula
- Driver
- PID
- Mode

- Simulation
- State
- Pump
- Resistance
- Constant
- Manual
- Range
- Any

ECO mode

On projectors that have **ECO** mode, special handling is required to wake up the projector. To wake up a projector that is in **ECO** mode:

- Send a wake on LAN request supplying the projectors HW (MAC) address
- Use the power button on the remote control
- Use the power button on the keypad
- Send a special command on the RS232 serial port

In the last case, waking up using the serial port, send the following ASCII characters:

:POWR1\r

Important note about the API documentation

Parts of the API are dynamic, other parts depend on peripherals or other factors. This means that the documentation shown here may not be complete with respect to a specific projector with a specific configuration. For example, if a lens is mounted that does not have motorized zoom, that part of the API will not be available, even if it's shown here. Another example is DMX. In its basic mode, only 2 channels are present. Setting it in extended mode will expose more channels, which may not be shown in this document.

The best way to know the exact API of your projector is to do an introspection as described in the previous chapters

Properties

Alphabetical list of all properties

dmx.artnet

Artnet enabled or not



Access: RW

Name	Type
artnet	hool

dmx.artnetnet

Artnet net selection



Access: RW

Name	Type
artnetnet	int

dmx.artnetuniverse

Artnet universe selection



Access: RW

Name	Туре
artnetuniverse	int

dmx.mode

Current mode



Access: RW

Name	Type
mode	string

dmx.monitor.channel01.function

Decription for the dmx channel



Name	Туре
function	strina

dmx.monitor.channel01.offset

Offset of the channel.



Access: R

Name	Type
offset	int

dmx.monitor.channel01.value

Current dmx value for the channel



Access: R

Name	Type
value	int

dmx.monitor.channel02.function

Decription for the dmx channel



Access: R

Name	Type
function	strina

dmx.monitor.channel02.offset

Offset of the channel.



Access: R

Name	Type
offset	int

dmx.monitor.channel02.value

Current dmx value for the channel



Name	Type
value	int

dmx.monitor.channel03.function

Decription for the dmx channel

MODELS UDX-4K22 UDX-W32

Access: R

Name Type function string

dmx.monitor.channel03.offset

Offset of the channel.

MODELS UDX-4K22 UDX-W32

Access: R

Name Type offset int

dmx.monitor.channel03.value

Current dmx value for the channel

MODELS UDX-4K22 UDX-W32

Access: R

Name Type value int

dmx.monitor.channel04.function

Decription for the dmx channel

MODELS UDX-4K22 UDX-W32

Access: R

Name Type function string

dmx.monitor.channel04.offset

Offset of the channel.

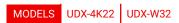
MODELS UDX-4K22 UDX-W32

Access: R

Name Type offset int

dmx.monitor.channel04.value

Current dmx value for the channel



Access: R

Name	Type
value	int

dmx.monitor.channel05.function

Decription for the dmx channel



Access: R

Name	Туре
function	strina

dmx.monitor.channel05.offset

Offset of the channel.



Access: R

Name	Type
offset	int

dmx.monitor.channel05.value

Current dmx value for the channel



Access: R

Name	Type
value	int

dmx.monitor.channel06.function

Decription for the dmx channel



Access: R

Name	Type
function	string

dmx.monitor.channel06.offset

Offset of the channel.

Access: R

Name	Type
offset	int

dmx.monitor.channel06.value

Current dmx value for the channel



Access: R

Name	Type
value	int

dmx.monitor.channel07.function

Decription for the dmx channel



Access: R

Name	Type
function	strina

dmx.monitor.channel07.offset

Offset of the channel.



Access: R

Name	Type
offset	int

dmx.monitor.channel07.value

Current dmx value for the channel



Access: R

Name	Type
value	int

dmx.monitor.channel08.function

Decription for the dmx channel



Name	Type
function	strina

dmx.monitor.channel08.offset

Offset of the channel.

MODELS UDX-4K22 UDX-W32

Access: R

Name Type offset int

dmx.monitor.channel08.value

Current dmx value for the channel

MODELS UDX-4K22 UDX-W32

Access: R

Name Type value int

dmx.monitor.channel09.function

Decription for the dmx channel

MODELS UDX-4K22 UDX-W32

Access: R

Name Type function string

dmx.monitor.channel09.offset

Offset of the channel.

MODELS UDX-4K22 UDX-W32

Access: R

Name Type offset int

dmx.monitor.channel09.value

Current dmx value for the channel

MODELS UDX-4K22 UDX-W32

Access: R

Name Type value int

dmx.monitor.channel10.function

Decription for the dmx channel

MODELS UDX-4K22 UDX-W32

Access: R

Name Type function string

dmx.monitor.channel10.offset

Offset of the channel.

MODELS UDX-4K22 UDX-W32

Access: R

Name Type offset int

dmx.monitor.channel10.value

Current dmx value for the channel

MODELS UDX-4K22 UDX-W32

Access: R

Name Type value int

dmx.monitor.channel11.function

Decription for the dmx channel

MODELS UDX-4K22 UDX-W32

Access: R

Name Type function string

dmx.monitor.channel11.offset

Offset of the channel.

MODELS UDX-4K22 UDX-W32

Access: R

Name Type offset int

dmx.monitor.channel11.value

Current dmx value for the channel



Access: R

Name	Type
value	int

dmx.monitor.channel12.function

Decription for the dmx channel

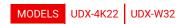


Access: R

Name	Туре
function	strina

dmx.monitor.channel12.offset

Offset of the channel.



Access: R

Name	Type	
offset	int	

dmx.monitor.channel12.value

Current dmx value for the channel



Access: R

Name	Type	
value	int	

dmx.monitor.channel13.function

Decription for the dmx channel



Access: R

Name	Type	
function	string	

dmx.monitor.channel13.offset

Offset of the channel.

Access: R

Name	Type	
offset	int	

dmx.monitor.channel13.value

Current dmx value for the channel



Access: R

Name	Type	
value	int	

dmx.monitor.channel14.function

Decription for the dmx channel



Access: R

Name	Туре
function	strina

dmx.monitor.channel14.offset

Offset of the channel.



Access: R

Name	Type	
offset	int	

dmx.monitor.channel14.value

Current dmx value for the channel



Access: R

Name	Type
value	int

dmx.monitor.connectionstate.active

true indicates that a dmx (if artnet setting is deactivated) or artnet package (if artnet setting is active) was received in the last 10 seconds.



Name	Type
active	bool

dmx.shutdown

Shutdown enabled or not



Access: RW

Name	Туре
shutdown	bool

dmx.shutdowntimeout

Time out for shutdown in minutes.



Access: RW

Name	Type
shutdowntimeout	int

dmx.startchannel

The dmx start channel [1..512].



Access: RW

Name	Type
startchannel	int

environment.alarmstate

Alarm state



Access: R

Name	Туре
alarmstate	enum
	Values
,	"Fatal"
	"Error"
	"Alert"
	"Warning"
	"0k"

firmware.firmwareversion

The version of the currently installed firmware.



Access: R

Name	Type	
firmwareversion	string	

gsm.available

The GSM card is present.



Access: R

Name	Type	
available	hool	

gsm.pin

Pin code for sim card.



_...

Access: RW

Name	Туре
pin	string

gsm.pinstate

The current state of PIN



Access: R

Name	Туре
pinstate	enum
	Values
	"Accepted"
	"Failed"
	"Locked"
	"Unknown"

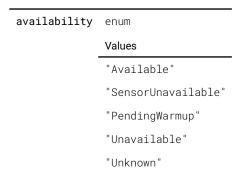
illumination.clo.availability

Shows the current availability.



Access: R

Name Type



illumination.clo.enable

True if constant light output is enabled, false otherwise

MODELS All

Access: RW

Name Type enable bool

illumination.clo.scale

The percentage to scale the setpoint by.

MODELS All

Access: RW

Name Type scale float

illumination.clo.setpoint

The target luminosity of the light source

MODELS All

Access: RW

Name Type setpoint float

illumination.clo.state

State of the CLO



Name	Туре
state	enum
	Values
	"0k"
	"TooDim"

"TooBright"

illumination.sources.laser.ispowerlimited

Whether power is currently limited.



Access: R

Name	Type	
ispowerlimited	bool	

illumination.sources.laser.maxpower

maximum power in percent



Access: R

Name	Туре
maxpower	float

illumination.sources.laser.minpower

minimum power in percent



Access: R

Name	Type
minnower	float

illumination.sources.laser.power

target power in percent



Access: RW

Name	Туре
power	float

illumination.sources.laser.powerlimitreason

If power is limited, gives the reason



Name	Туре	
powerlimitreason	string	

illumination.state

The state of light



Access: R

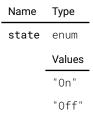


image.blackcontentdetection.dimminginterval

DEPRECATED: Interval in milliseconds during which the power dimming is done upon black content detection.



Access: RW

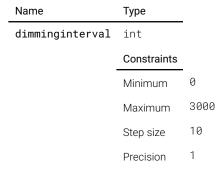


image.blackcontentdetection.enable

DEPRECATED: Enable/disable black content detection.



Access: RW



image. black content detection. sample interval

DEPRECATED: Sample interval in milliseconds for black content detection.



Name	Туре	
sampleinterval	int	
	Constraints	
	Minimum	500

Maximum 3000
Step size 10
Precision 1

image.blackcontentdetection.state

DEPRECATED: State true/false means black content was/was not detected.



Access: R

Name	Type
state	bool

image.blackcontentdetection.threshold

DEPRECATED: The offset seen from the signal range lowest value, which specifies the black level threshold for the detector.



Access: RW

Name	Type	
threshold	int	
	Constraints	
	Minimum	0
	Maximum	255
	Step size	1
	Precision	1

image.brightness

Image brightness/offset. The value is normalized, 0 is default, 1 is 100% offset.



Access: RW

Name	Туре	
brightness	float	
	Constraints	
	Minimum	-1
	Maximum	1
	Step size	1
	Precision	0.01

image.color.p7.custom.bluegain

Desired blue gain value

MODELS All

Access: RW

Name	Type
hluegain	float

image.color.p7.custom.bluelum

Desired blue luminanace

MODELS All

Access: RW

Name	Type
bluelum	float

image.color.p7.custom.bluex

Desired blue x in xy-coordinates

MODELS All

Access: RW

Name	Type
bluex	float

image.color.p7.custom.bluey

Desired blue y in xy-coordinates

MODELS All

Access: RW

Name	Type
bluev	float

image.color.p7.custom.cmyreadable

true if secondaries should be shown (OSD)

MODELS All

Access: R

Name	Type
cmyreadable	bool

image.color.p7.custom.cmywritable

true if secondaries are Writable



Name	Type
cmywritable	bool

image.color.p7.custom.cyangain

Desired cyan gain value



Access: RW

Name	Туре	
cyangain	float	

image.color.p7.custom.cyanlum

Desired cyan luminanace



Access: RW

Name	Туре
cyanlum	float

image.color.p7.custom.cyanx

Desired cyan x in xy-coordinates



Access: RW

Name	Туре
cyanx	float

image.color.p7.custom.cyany

Desired cyan y in xy-coordinates



Access: RW

Name	Туре
cyany	float

image.color.p7.custom.gainsavailable

true when gains are available



Name	Type
nainsavailahle	hool

image.color.p7.custom.greengain

Desired green gain value



Access: RW

Name	Туре
greengain	float

image.color.p7.custom.greenlum

Desired green luminanace



Access: RW

Name	Туре
greenlum	float

image.color.p7.custom.greenx

Desired green x in xy-coordinates



Access: RW

Name	Туре
greenx	float

image.color.p7.custom.greeny

Desired green y in xy-coordinates



Access: RW

Name	Туре
greeny	float

image.color.p7.custom.luminancesavailable

true if luminances are available



Access: R

Name	Type
luminancesavailable	bool

image.color.p7.custom.magentagain

Desired magenta gain value



Access: RW

Name	Type
magentagain	float

image.color.p7.custom.magentalum

Desired magenta luminanace



Access: RW

Name	Туре
magentalum	float

image.color.p7.custom.magentax

Desired magenta x in xy-coordinates



Access: RW

Name	Туре
magentax	float

image.color.p7.custom.magentay

Desired magenta y in xy-coordinates



Access: RW

Name	Туре
magentav	float

image.color.p7.custom.mode

Description not provided



Access: RW

Name	Туре
mode	string

image.color.p7.custom.modes

Description not provided



Access: R

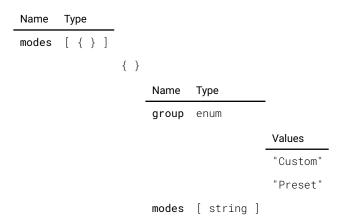


image.color.p7.custom.redgain

Desired red gain value



Access: RW

Name	Type
redgain	float

image.color.p7.custom.redlum

Desired red luminanace

MODELS All

Access: RW

Name	Туре
redlum	float

image.color.p7.custom.redx

Desired red x in xy-coordinates



Access: RW

Name	Туре
redx	float

image.color.p7.custom.redy

Desired red y in xy-coordinates



Name	Type
redy	float

image.color.p7.custom.rgbcmygainsavailable

true when R,G,B,C,M,Y gains are available



Access: R

Name	Туре
rgbcmygainsavailable	bool

image.color.p7.custom.rgbwritable

true if RGB are Writable (not in WHITE mode)



Access: R

image.color.p7.custom.target

Target color coordinates for the current preset

MODELS All

Access: R

image.color.p7.custom.whitegain

У

Name

white { }

float

Type float float Desired white gain value



Access: RW

Name	Туре
whitegain	float

image.color.p7.custom.whitegain available

true when white gain is available



Access: R

Name	Type
whitegainavailable	bool

image.color.p7.custom.whitelum

Desired white luminanace



Access: RW

Name	Туре
whitelum	float

image.color.p7.custom.whitemode

Description not provided



Access: RW

Name	Туре
whitemode	enum
	Values
	"Coordinates"
	"Temperature"

image.color.p7.custom.whitetemperature

Desired white point temperature



Name	Туре
whitetemperature	int
	Constraints

Minimum 3200

Maximum 13000

Step size 100

Precision 1

image.color.p7.custom.whitetemperatureavailable

true if White temperature is available



Access: R

Name	Туре
whitetemperatureavailable	bool

image.color.p7.custom.whitewritable

true if White is Writable



Access: R

Name	Type	
whitewritable	bool	

image.color.p7.custom.whitex

Desired white x in xy-coordinates



Access: RW

Name	Туре
whitex	float

image.color.p7.custom.whitey

Desired white y in xy-coordinates



Access: RW

Name	Туре
whitey	float

image.color.p7.custom.yellowgain

Desired yellow gain value



Name	Туре
yellowgain	float

image.color.p7.custom.yellowlum

Desired yellow luminanace



Access: RW

Name	Туре
yellowlum	float

image.color.p7.custom.yellowx

Desired yellow x in xy-coordinates



Access: RW

Name	Туре
yellowx	float

image.color.p7.custom.yellowy

Desired yellow y in xy-coordinates



Access: RW

Name	Type	
yellowy	float	

image.color.p7.native.c1

Native C1 x in xy-coordinates



Access: R

Name	Туре	
c1	{ }	
	Name	Туре
	х	float
	у	float
	lum	float

image.color.p7.native.c1available

Description not provided



Access: R

Name	Type
c1available	hool

image.color.p7.native.c2

Native C2 x in xy-coordinates



Access: R

Name	Туре	
c2	{ }	
	Name	Туре
	х	float
	у	float
	lum	float

image.color.p7.native.c2available

Description not provided



Access: R

Name	Туре
c2available	bool

image.color.p7.native.list

list available native sets



Access: R

Name	Туре	
list	string	1

image.color.p7.native.normal.c1

Native C1 x in xy-coordinates



Name	Туре	
c1	{ }	
	Name	Туре
	х	float
	у	float

lum float

image.color.p7.native.normal.c1available

Description not provided



Access: R

Name	Type
c1available	hool

image.color.p7.native.normal.c2

Native C2 x in xy-coordinates



Access: R



Name	Type
Х	float
у	float
lum	float

image.color.p7.native.normal.c2available

Description not provided



. . .

Access: R

Name	Type	
c2available	bool	

image.color.p7.native.normal.rgbw

Native red x in xy-coordinates





Name	Туре
Х	float
V	float

lum float ${\tt green} \quad \{ \ \ \}$ Name Туре float Х float у lum float blue { } Name Type float float у float lum $\quad \text{white} \quad \{\ \}$ Name Type float float у lum float

image.color.p7.native.rgbw

Native red x in xy-coordinates

MODELS All

Access: R

Name	Туре			
rgbw	{ }			
	Name	Туре	e	
	red	{ }	_	
			Name	Туре
			х	floa
			у	floa
			lum	floa
	green	{ }		
			Name	Type

Туре
float
float
float

float float

float

 $\quad \text{blue} \quad \{\ \}$

Name	Type
Х	float
у	float
lum	float

 $\quad \text{white} \quad \{ \ \ \}$

X	floa
y	floa
lum	float

image.color.rgbmode.rgbmode

RGB Mode



Access: RW

	_
Name	Туре
rgbmode	enum
	Values
	"Full"
	"Red"
	"Green"
	"Blue"
	"RedGreen"
	"GreenBlue"
	"BlueRed"

image.connector.l1displayport.colorprimaries

Override the detected signal color primaries. Set to Auto for automatic control.



Access: RW

Name	Туре
colorprimaries	enum
	Values
	"Auto"
	"Uncorrected"
	"REC709"
	"REC2020"
	"DCI-P3-D65"
	"DCI-P3-Theater"

image.connector. I 1 display port. color prima rie savailable

true if Color Primaries is available



Name	Туре
colorprimariesavailable	hool

image.connector.l1displayport.colorspace

Override the detected signal color space. Set to Auto for automatic control.



Access: RW

Name	Туре
colorspace	enum
	Values
,	"Auto"
	"RGB"
	"REC709"
	"REC601"
	"REC2020"

image.connector.l1displayport.detectedsignal

The signal information of the currently detected signal. If 'active' is false, there is no detected signal and the rest of the information should be disregarded. is_stereo indicates if stereo_mode is different from none.



Name	Туре
detectedsignal	{ }

Name	Type
active	bool
name	string
vertical_total	int
horizontal_total	int
vertical_resolution	int
horizontal_resolution	int
vertical_sync_width	int
vertical_front_porch	int
vertical_back_porch	int
horizontal_sync_width	int
horizontal_front_porch	int
horizontal_back_porch	int
horizontal_frequency	float
vertical_frequency	float
pixel_rate	int
scan	enum

Values	

[&]quot;Progressive"

hits per component	int	
<pre>bits_per_component color_space</pre>	enum	
color_space	eriulli	Values
		"RGB"
		"REC709"
		"REC601"
		"REC2020"
signal_range	enum	
		Values
		"0-255"
		"16-235"
chroma_sampling	enum	
		Values
		"4:4:4"
		"4:2:2"
		"4:2:0"
gamma_type	enum	
		Values
		"POWER"
		"sRGB"
		"REC_BT1886"
		"SMPTE_ST2084"
color_primaries	enum	
coror_primarites	CHain	
cotor_primaries	Cridiii	Values
COTOT_pt Imai Tes	Crium	Values "REC709"
COTOT_pt Imai Tes	Citam	
COTOT_pt Imai Tes	Citam	"REC709"
COTOT_pt Imai Tes	Citum	"REC709" "REC2020"
mastering_luminance	float	"REC709" "REC2020" "DCI-P3-D65"
		"REC709" "REC2020" "DCI-P3-D65"
mastering_luminance	float	"REC709" "REC2020" "DCI-P3-D65"
mastering_luminance	float	"REC709" "REC2020" "DCI-P3-D65" "DCI-P3-Theater"
mastering_luminance	float	"REC709" "REC2020" "DCI-P3-D65" "DCI-P3-Theater"
mastering_luminance	float	"REC709" "REC2020" "DCI-P3-D65" "DCI-P3-Theater" Values
mastering_luminance	float	"REC709" "REC2020" "DCI-P3-D65" "DCI-P3-Theater" Values "5:4" "4:3"
mastering_luminance	float	"REC709" "REC2020" "DCI-P3-D65" "DCI-P3-Theater" Values "5:4" "4:3" "16:10"
mastering_luminance	float	"REC709" "REC2020" "DCI-P3-D65" "DCI-P3-Theater" Values "5:4" "4:3" "16:10" "16:9"
mastering_luminance	float	"REC709" "REC2020" "DCI-P3-D65" "DCI-P3-Theater" Values "5:4" "4:3" "16:10" "16:9" "1.85:1"
mastering_luminance	float	"REC709" "REC2020" "DCI-P3-D65" "DCI-P3-Theater" Values "5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1"
mastering_luminance	float	"REC709" "REC2020" "DCI-P3-D65" "DCI-P3-Theater" Values "5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1"
mastering_luminance	float	"REC709" "REC2020" "DCI-P3-D65" "DCI-P3-Theater" Values "5:4" "4:3" "16:10" "16:9" "1.85:1" "2.35:1" "2.37:1"
mastering_luminance	float	"REC709" "REC2020" "DCI-P3-D65" "DCI-P3-Theater" Values "5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
mastering_luminance content_aspect_ratio	float	"REC709" "REC2020" "DCI-P3-D65" "DCI-P3-Theater" Values "5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"

Values
"None"
"Sequential"
"FramePacked"
"TopBottom"
"SideBySide"

image.connector.l1displayport.edid.selected

Selected EDID for connector



Access: RW

Name	Туре
selected	string

image.connector.l1displayport.signalrange

Override the detected signal signal range. Set to Auto for automatic control.



Access: RW

Name	Туре
signalrange	enum
	Values
	"Auto"
	"0-255"
	"16-235"

image.connector.l1hdbaset1.colorprimaries

Override the detected signal color primaries. Set to Auto for automatic control.



Access: RW

Name	Туре
colorprimaries	enum
	Values
	"Auto"
	"Uncorrected"
	"REC709"
	"REC2020"
	"DCI-P3-D65"
	"DCI-P3-Theater"

image.connector.l1hdbaset1.colorprimariesavailable

true if Color Primaries is available



Access: R

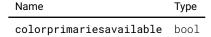


image.connector.l1hdbaset1.colorspace

Override the detected signal color space. Set to Auto for automatic control.



Access: RW

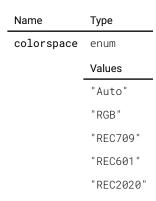


image.connector.l1hdbaset1.detectedsignal

The signal information of the currently detected signal. If 'active' is false, there is no detected signal and the rest of the information should be disregarded. is_stereo indicates if stereo_mode is different from none.



Name	Туре	
detectedsignal	{ }	
	Name	Туре
	active	bool
	name	string
	vertical_total	int
	horizontal_total	int
	vertical_resolution	int
	horizontal_resolution	int
	vertical_sync_width	int
	vertical_front_porch	int
	vertical_back_porch	int
	horizontal_sync_width	int
	horizontal_front_porch	int
	horizontal_back_porch	int

horizontal_frequency	float	
vertical_frequency	float	
pixel_rate	int	
scan	enum	
		Values
		"Progressive"
		"Interlaced"
bits_per_component	int	
color_space	enum	
		Values
		"RGB"
		"REC709"
		"REC601"
		"REC2020"
signal_range	enum	
		Values
		"0-255"
		"16-235"
chroma_sampling	enum	
. •		Values
		"4:4:4"
		"4:2:2"
		"4:2:0"
gamma_type	enum	1.2.0
gaa_c, po	o.ra	Values
		"POWER"
		"sRGB"
		"REC_BT1886"
		"SMPTE_ST2084"
1		3MF1E_312004
color_primaries	enum	
		Values "REC709"
		"REC2020"
		"DCI-P3-D65"
_		"DCI-P3-Theater"
mastering_luminance	float	
content_aspect_ratio	enum	
		Values
		"5:4"
		"4:3"
		"16:10"
		"16:9"
		"1.85:1"
		Reference Guide

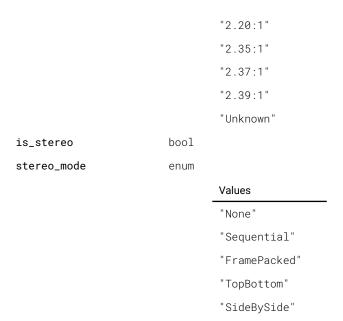


image.connector.l1hdbaset1.edid.selected

Selected EDID for connector



Access: RW

Name	Туре
selected	string

image.connector.l1hdbaset1.signalrange

Override the detected signal signal range. Set to Auto for automatic control.



Access: RW

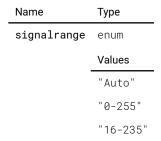
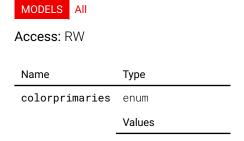


image.connector.l1hdbaset2.colorprimaries

Override the detected signal color primaries. Set to Auto for automatic control.



"Auto"

"Uncorrected"

"REC709"

"REC2020"

"DCI-P3-D65"

"DCI-P3-Theater"

image.connector.l1hdbaset2.colorprimariesavailable

true if Color Primaries is available



Access: R



image.connector.l1hdbaset2.colorspace

Override the detected signal color space. Set to Auto for automatic control.



Access: RW

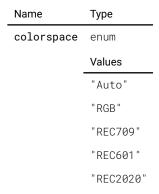


image.connector.l1hdbaset2.detectedsignal

The signal information of the currently detected signal. If 'active' is false, there is no detected signal and the rest of the information should be disregarded. is_stereo indicates if stereo_mode is different from none.



	Name	Туре	
_	detectedsignal	{ }	
		Name	Туре
		active	bool
		name	string
		vertical_total	int
		horizontal_total	int

vertical_resolution	int	
horizontal_resolution	int	
vertical_sync_width	int	
vertical_front_porch	int	
vertical_back_porch	int	
horizontal_sync_width	int	
horizontal_front_porch	int	
horizontal_back_porch	int	
horizontal_frequency	float	
vertical_frequency	float	
pixel_rate	int	
scan	enum	
		Values
		"Progressive"
		"Interlaced"
bits_per_component	int	
color_space	enum	
		Values
		"RGB"
		"REC709"
		"REC601"
		"REC2020"
signal_range	enum	
		Values
		"0-255"
		"16-235"
chroma_sampling	enum	
		Values
		"4:4:4"
		"4:2:2"
		"4:2:0"
gamma_type	enum	
		Values
		"POWER"
		"sRGB"
		"REC_BT1886"
		"SMPTE_ST2084"
color_primaries	enum	
		Values
		"REC709"
		"REC2020"
		"DCI-P3-D65"
		"DCI-P3-Theater"
		DOI . O THOUSEN

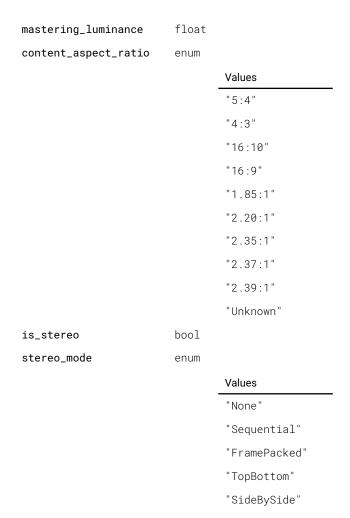


image.connector.l1hdbaset2.edid.selected

Selected EDID for connector



Access: RW

Name	Туре
selected	strina

image.connector.l1hdbaset2.signalrange

Override the detected signal signal range. Set to Auto for automatic control.



Access: RW

Name	Type
signalrange	enum
	Values
	"Auto"
	"0-255"
	"16-235"

image.connector.l1hdmi.colorprimaries

Override the detected signal color primaries. Set to Auto for automatic control.



Access: RW

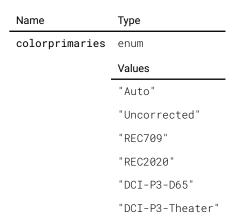


image.connector.l1hdmi.colorprimariesavailable

true if Color Primaries is available



Access: R

Name	Type
colorprimariesavailable	bool

image.connector.l1hdmi.colorspace

Override the detected signal color space. Set to Auto for automatic control.



Access: RW

Name	Туре
colorspace	enum
	Values
	"Auto"
	"RGB"
	"REC709"
	"REC601"
	"REC2020"

image.connector.l1hdmi.detectedsignal

The signal information of the currently detected signal. If 'active' is false, there is no detected signal and the rest of the information should be disregarded. is_stereo indicates if stereo_mode is different from none.



Name Type detectedsignal { }

Name	Туре
active	bool
name	string
vertical_total	int
horizontal_total	int
vertical_resolution	int
horizontal_resolution	int
vertical_sync_width	int
vertical_front_porch	int
vertical_back_porch	int
horizontal_sync_width	int
horizontal_front_porch	int
horizontal_back_porch	int
horizontal_frequency	float
vertical_frequency	float
pixel_rate	int
scan	enum

		Values
		"Progressive"
		"Interlaced"
bits_per_component	int	
color_space	enum	
		Values
		"RGB"
		"REC709"
		"REC601"
		"REC2020"
signal_range	enum	
		Values
		"0-255"
		"16-235"
chroma_sampling	enum	
		Values
		"4:4:4"
		"4:2:2"
		"4:2:0"

enum

 ${\tt gamma_type}$

"REC_BT1886"

Values
"POWER"
"sRGB"

"SMPTE_ST2084"

Values
"REC709"
"REC2020"
"DCI-P3-D65"
"DCI-P3-Theater"

mastering_luminance float
content_aspect_ratio enum

	Values
	"5:4"
	"4:3"
	"16:10"
	"16:9"
	"1.85:1"
	"2.20:1"
	"2.35:1"
	"2.37:1"
	"2.39:1"
	"Unknown"
bool	
enum	

"None"

"Sequential"

"FramePacked"

"TopBottom"

"SideBySide"

Values

image.connector.l1hdmi.edid.selected

is_stereo

 ${\tt stereo_mode}$

Selected EDID for connector



Access: RW

Name	Type	
selected	string	

image.connector.l1hdmi.signalrange

Override the detected signal signal range. Set to Auto for automatic control.



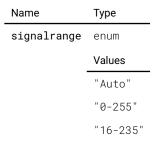


image.connector.l1sdia.colorprimaries

Override the detected signal color primaries. Set to Auto for automatic control.



Access: RW

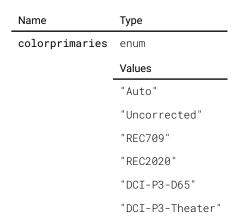


image. connector. I 1 s dia. color prima rie savailable

true if Color Primaries is available



Access: R

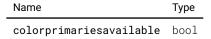


image.connector.l1sdia.colorspace

Override the detected signal color space. Set to Auto for automatic control.



Name	Туре	
colorspace	enum	
	Values	
	"Auto"	
	"RGB"	
	"REC709"	
	"REC601"	

"REC2020"

image.connector.l1sdia.detectedsignal

The signal information of the currently detected signal. If 'active' is false, there is no detected signal and the rest of the information should be disregarded. is_stereo indicates if stereo_mode is different from none.



Name	Туре		
detectedsignal	{ }	•	
	Name	Туре	
	active	bool	•
	name	string	
	vertical_total	int	
	horizontal_total	int	
	vertical_resolution	int	
	$horizontal_resolution$	int	
	vertical_sync_width	int	
	vertical_front_porch	int	
	vertical_back_porch	int	
	horizontal_sync_width	int	
	horizontal_front_porch	int	
	horizontal_back_porch	int	
	horizontal_frequency	float	
	vertical_frequency	float	
	pixel_rate	int	
	scan	enum	
			Values
			"Progressive"
			"Interlaced"
	bits_per_component	int	
	color_space	enum	
			Values
			"RGB"
			"REC709"
			"REC601"
			"REC2020"
	signal_range	enum	
			Values
			"0-255"
			"16-235"
	chroma_sampling	enum	

		Values
	•	"4:4:4"
		"4:2:2"
		"4:2:0"
gamma_type	enum	
		Values
	•	"POWER"
		"sRGB"
		"REC_BT1886"
		"SMPTE_ST2084"
color_primaries	enum	
		Values
	•	"REC709"
		"REC2020"
		"DCI-P3-D65"
		"DCI-P3-Theater"
mastering_luminance	float	
content_aspect_ratio	enum	
		Values
		Values
	-	"5:4"
	•	
	-	"5:4"
	-	"5:4" "4:3"
		"5:4" "4:3" "16:10"
		"5:4" "4:3" "16:10" "16:9"
	•	"5:4" "4:3" "16:10" "16:9" "1.85:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1"
is_stereo	bool	"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
is_stereo stereo_mode	bool enum	"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"

image.connector.l1sdia.signalrange

Override the detected signal signal range. Set to Auto for automatic control.



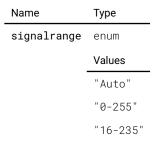


image.connector.l1sdib.colorprimaries

Override the detected signal color primaries. Set to Auto for automatic control.



Access: RW

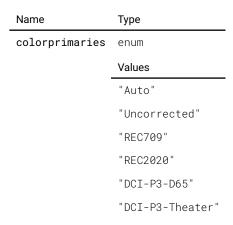


image.connector.l1sdib.colorprimariesavailable

true if Color Primaries is available



Access: R

Name	Туре
colorprimariesavailable	bool

image.connector.l1sdib.colorspace

Override the detected signal color space. Set to Auto for automatic control.



Name	Туре	
colorspace	enum	
	Values	
	"Auto"	
	"RGB"	
	"REC709"	
	"REC601"	

"REC2020"

image.connector.l1sdib.detectedsignal

The signal information of the currently detected signal. If 'active' is false, there is no detected signal and the rest of the information should be disregarded. is_stereo indicates if stereo_mode is different from none.



Name	Туре		
detectedsignal	{ }	•	
	Name	Туре	
	active	bool	•
	name	string	
	vertical_total	int	
	horizontal_total	int	
	vertical_resolution	int	
	$horizontal_resolution$	int	
	vertical_sync_width	int	
	vertical_front_porch	int	
	vertical_back_porch	int	
	horizontal_sync_width	int	
	horizontal_front_porch	int	
	horizontal_back_porch	int	
	horizontal_frequency	float	
	vertical_frequency	float	
	pixel_rate	int	
	scan	enum	
			Values
			"Progressive"
			"Interlaced"
	bits_per_component	int	
	color_space	enum	
			Values
			"RGB"
			"REC709"
			"REC601"
			"REC2020"
	signal_range	enum	
			Values
			"0-255"
			"16-235"
	chroma_sampling	enum	

		Values
		"4:4:4"
		"4:2:2"
		"4:2:0"
gamma_type	enum	
		Values
		"POWER"
		"sRGB"
		"REC_BT1886"
		"SMPTE_ST2084"
color_primaries	enum	
		Values
		"REC709"
		"REC2020"
		"DCI-P3-D65"
		"DCI-P3-Theater"
mastering_luminance	float	
content_aspect_ratio	enum	
		Values
		Values " 5 : 4 "
		"5:4"
		"5:4"
		"5:4" "4:3" "16:10"
		"5:4" "4:3" "16:10" "16:9"
		"5:4" "4:3" "16:10" "16:9" "1.85:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1"
is_stereo	bool	"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
<pre>is_stereo stereo_mode</pre>	bool enum	"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.37:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"

image.connector.l1sdib.signalrange

Override the detected signal signal range. Set to Auto for automatic control.



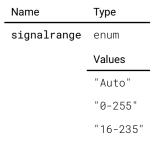


image.connector.l1sdic.colorprimaries

Override the detected signal color primaries. Set to Auto for automatic control.



Access: RW

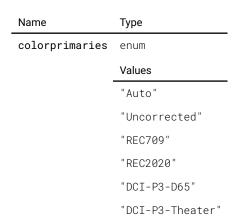


image.connector.l1sdic.colorprimariesavailable

true if Color Primaries is available



Access: R

Name	Type
colorprimariesavailable	bool

image.connector.l1sdic.colorspace

Override the detected signal color space. Set to Auto for automatic control.



Name	Туре	
colorspace	enum	
	Values	
	"Auto"	
	"RGB"	
	"REC709"	
	"REC601"	

"REC2020"

image.connector.l1sdic.detectedsignal

The signal information of the currently detected signal. If 'active' is false, there is no detected signal and the rest of the information should be disregarded. is_stereo indicates if stereo_mode is different from none.



Name	Туре		
detectedsignal	{ }	•	
	Name	Туре	
	active	bool	•
	name	string	
	vertical_total	int	
	horizontal_total	int	
	vertical_resolution	int	
	$horizontal_resolution$	int	
	vertical_sync_width	int	
	vertical_front_porch	int	
	vertical_back_porch	int	
	horizontal_sync_width	int	
	horizontal_front_porch	int	
	horizontal_back_porch	int	
	horizontal_frequency	float	
	vertical_frequency	float	
	pixel_rate	int	
	scan	enum	
			Values
			"Progressive"
			"Interlaced"
	bits_per_component	int	
	color_space	enum	
			Values
			"RGB"
			"REC709"
			"REC601"
			"REC2020"
	signal_range	enum	
			Values
			"0-255"
			"16-235"
	chroma_sampling	enum	

		Values
	•	"4:4:4"
		"4:2:2"
		"4:2:0"
gamma_type	enum	
		Values
	•	"POWER"
		"sRGB"
		"REC_BT1886"
		"SMPTE_ST2084"
color_primaries	enum	
		Values
	•	"REC709"
		"REC2020"
		"DCI-P3-D65"
		"DCI-P3-Theater"
mastering_luminance	float	
content_aspect_ratio	enum	
		Values
		Values
	-	"5:4"
	•	
	-	"5:4"
	-	"5:4" "4:3"
		"5:4" "4:3" "16:10"
		"5:4" "4:3" "16:10" "16:9"
	•	"5:4" "4:3" "16:10" "16:9" "1.85:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1"
is_stereo	bool	"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
is_stereo stereo_mode	bool enum	"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"

image.connector.l1sdic.signalrange

Override the detected signal signal range. Set to Auto for automatic control.



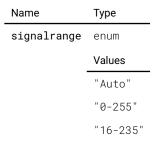


image.connector.l1sdid.colorprimaries

Override the detected signal color primaries. Set to Auto for automatic control.



Access: RW

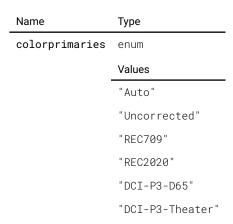


image.connector.l1sdid.colorprimariesavailable

true if Color Primaries is available



Access: R

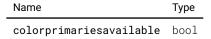


image.connector.l1sdid.colorspace

Override the detected signal color space. Set to Auto for automatic control.



Name	Туре
colorspace	enum
	Values
	"Auto"
	"RGB"
	"REC709"
	"REC601"

"REC2020"

image.connector.l1sdid.detectedsignal

The signal information of the currently detected signal. If 'active' is false, there is no detected signal and the rest of the information should be disregarded. is_stereo indicates if stereo_mode is different from none.



Name	Туре		
detectedsignal	{ }	•	
	Name	Туре	
	active	bool	•
	name	string	
	vertical_total	int	
	horizontal_total	int	
	vertical_resolution	int	
	horizontal_resolution	int	
	vertical_sync_width	int	
	vertical_front_porch	int	
	vertical_back_porch	int	
	horizontal_sync_width	int	
	horizontal_front_porch	int	
	horizontal_back_porch	int	
	horizontal_frequency	float	
	vertical_frequency	float	
	pixel_rate	int	
	scan	enum	
			Values
			"Progressive"
			"Interlaced"
	bits_per_component	int	
	color_space	enum	
			Values
			"RGB"
			"REC709"
			"REC601"
			"REC2020"
	signal_range	enum	
			Values
			"0-255"
			"16-235"
	chroma_sampling	enum	

		Values
		"4:4:4"
		"4:2:2"
		"4:2:0"
gamma_type	enum	
		Values
		"POWER"
		"sRGB"
		"REC_BT1886"
		"SMPTE_ST2084"
color_primaries	enum	
		Values
		"REC709"
		"REC2020"
		"DCI-P3-D65"
		"DCI-P3-Theater"
mastering_luminance	float	
content_aspect_ratio	enum	
		Values
		Values " 5 : 4 "
		"5:4"
		"5:4"
		"5:4" "4:3" "16:10"
		"5:4" "4:3" "16:10" "16:9"
		"5:4" "4:3" "16:10" "16:9" "1.85:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1"
is_stereo	bool	"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
<pre>is_stereo stereo_mode</pre>	bool enum	"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.37:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"

image.connector.l1sdid.signalrange

Override the detected signal signal range. Set to Auto for automatic control.



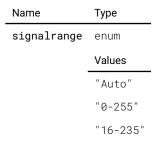


image.connector.l2displayporta.colorprimaries

Override the detected signal color primaries. Set to Auto for automatic control.



Access: RW

Name	Туре
colorprimaries	enum
	Values
	"Auto"
	"Uncorrected"
	"REC709"
	"REC2020"
	"DCI-P3-D65"
	"DCI-P3-Theater"

image.connector.l2displayporta.colorprimariesavailable

true if Color Primaries is available



Access: R

Name	Type
colorprimariesavailable	bool

image.connector.l2displayporta.colorspace

Override the detected signal color space. Set to Auto for automatic control.



Name	Туре
colorspace	enum
	Values
	"Auto"
	"RGB"
	"REC709"
	"REC601"

"REC2020"

image.connector.l2displayporta.detectedsignal

The signal information of the currently detected signal. If 'active' is false, there is no detected signal and the rest of the information should be disregarded. is_stereo indicates if stereo_mode is different from none.



Name	Туре		
detectedsignal	{ }	•	
	Name	Туре	
	active	bool	•
	name	string	
	vertical_total	int	
	horizontal_total	int	
	vertical_resolution	int	
	horizontal_resolution	int	
	vertical_sync_width	int	
	vertical_front_porch	int	
	vertical_back_porch	int	
	horizontal_sync_width	int	
	horizontal_front_porch	int	
	horizontal_back_porch	int	
	horizontal_frequency	float	
	vertical_frequency	float	
	pixel_rate	int	
	scan	enum	
			Values
			"Progressive"
			"Interlaced"
	bits_per_component	int	
	color_space	enum	
			Values
			"RGB"
			"REC709"
			"REC601"
			"REC2020"
	signal_range	enum	
			Values
			"0-255"
			"16-235"
	chroma_sampling	enum	

		Values
		"4:4:4"
		"4:2:2"
		"4:2:0"
gamma_type	enum	
		Values
		"POWER"
		"sRGB"
		"REC_BT1886"
		"SMPTE_ST2084"
color_primaries	enum	
		Values
		"REC709"
		"REC2020"
		"DCI-P3-D65"
		"DCI-P3-Theater"
mastering_luminance	float	
content_aspect_ratio	enum	
		Values
		Values "5:4"
		"5:4"
		"5:4"
		"5:4" "4:3" "16:10"
		"5:4" "4:3" "16:10" "16:9"
		"5:4" "4:3" "16:10" "16:9" "1.85:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1"
is_stereo	bool	"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
<pre>is_stereo stereo_mode</pre>	bool enum	"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"

image.connector.l2displayporta.edid.selected

Selected EDID for connector



Name	Type
selected	string

image.connector.l2displayporta.signalrange

Override the detected signal signal range. Set to Auto for automatic control.



Access: RW

Name	Туре
signalrange	enum
	Values
	"Auto"
	"0-255"
	"16-235"

image.connector.l2displayportb.colorprimaries

Override the detected signal color primaries. Set to Auto for automatic control.



Access: RW

Name	Туре
colorprimaries	enum
	Values
	"Auto"
	"Uncorrected"
	"REC709"
	"REC2020"
	"DCI-P3-D65"
	"DCI-P3-Theater"

image.connector.l2displayportb.colorprimariesavailable

true if Color Primaries is available



Name Type colorprimariesavailable bool

image.connector.l2displayportb.colorspace

Override the detected signal color space. Set to Auto for automatic control.

MODELS UDX-4K32 UDX-4K22

Access: RW

Name	Туре
colorspace	enum
	Values
,	"Auto"
	"RGB"
	"REC709"
	"REC601"
	"REC2020"

image.connector.l2displayportb.detectedsignal

The signal information of the currently detected signal. If 'active' is false, there is no detected signal and the rest of the information should be disregarded. is_stereo indicates if stereo_mode is different from none.



Name	Туре
detectedsignal	{ }

Name	Туре
active	bool
name	string
vertical_total	int
horizontal_total	int
vertical_resolution	int
$horizontal_resolution$	int
vertical_sync_width	int
vertical_front_porch	int
vertical_back_porch	int
horizontal_sync_width	int
horizontal_front_porch	int
horizontal_back_porch	int
horizontal_frequency	float
vertical_frequency	float
pixel_rate	int
scan	enum

scan	enum	
		Values
		"Progressive"
		"Interlaced"
bits_per_component	int	
color_space	enum	
		Values
		"RGB"

		Nererence duide
		"REC709"
		"REC601"
		"REC2020"
signal_range	enum	
		Values
		"0-255"
		"16-235"
chroma_sampling	enum	
- , 3		Values
		"4:4:4"
		"4:2:2"
		"4:2:0"
gamma typa	onum	4.2.0
gamma_type	enum	
		Values
		"POWER"
		"sRGB"
		"REC_BT1886"
		"SMPTE_ST2084"
color_primaries	enum	
		Values
		"REC709"
		"REC2020"
		"DCI-P3-D65"
		"DCI-P3-Theater"
mastering_luminance	float	
content_aspect_ratio	enum	
		Values
		Values "5:4"
		"5:4"
		"5:4" "4:3"
		"5:4" "4:3" "16:10"
		"5:4" "4:3" "16:10" "16:9"
		"5:4" "4:3" "16:10" "16:9" "1.85:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1"
is_stereo	bool	"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1"
<pre>is_stereo stereo_mode</pre>		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1"
<pre>is_stereo stereo_mode</pre>	bool enum	"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.37:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"
		"5:4" "4:3" "16:10" "16:9" "1.85:1" "2.20:1" "2.35:1" "2.37:1" "2.39:1" "Unknown"

"TopBottom"

"SideBySide"

image.connector.l2displayportb.edid.selected

Selected EDID for connector



Access: RW

Name	Туре
selected	string

image.connector.l2displayportb.signalrange

Override the detected signal signal range. Set to Auto for automatic control.

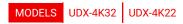


Access: RW

Name	Туре
signalrange	enum
	Values
	"Auto"
	"0-255"
	"16-235"

image.connector.l2displayportc.colorprimaries

Override the detected signal color primaries. Set to Auto for automatic control.



Access: RW

Name	Туре	
colorprimaries	enum	
	Values	
	"Auto"	
	"Uncorrected"	
	"REC709"	
	"REC2020"	
	"DCI-P3-D65"	
	"DCI-P3-Theater"	

image.connector. I2 display portc. color primaries available

true if Color Primaries is available



Access: R

Name	Type
colorprimariesavailable	hool

image.connector.l2displayportc.colorspace

Override the detected signal color space. Set to Auto for automatic control.



Access: RW

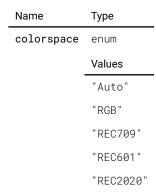


image.connector.l2displayportc.detectedsignal

The signal information of the currently detected signal. If 'active' is false, there is no detected signal and the rest of the information should be disregarded. is_stereo indicates if stereo_mode is different from none.

MODELS UDX-4K32 UDX-4K22

Name	Туре
detectedsignal	<i>f</i> \

Name	Туре
active	bool
name	string
vertical_total	int
horizontal_total	int
vertical_resolution	int
horizontal_resolution	int
vertical_sync_width	int
vertical_front_porch	int
vertical_back_porch	int
horizontal_sync_width	int
$\verb horizontal_front_porch $	int
horizontal_back_porch	int
horizontal_frequency	float
vertical_frequency	float
pixel_rate	int

0000	ODLIM	
scan	enum	Values
		"Progressive"
		"Interlaced"
bits_per_component	int	inter faced
color_space	enum	
00101 <u>-</u> 00400	craiii	Values
		"RGB"
		"REC709"
		"REC601"
		"REC2020"
signal_range	enum	
3 - 2 - 3		Values
		"0-255"
		"16-235"
chroma_sampling	enum	
3		Values
		"4:4:4"
		"4:2:2"
		"4:2:0"
gamma_type	enum	
J.		Values
		"POWER"
		"sRGB"
		"REC_BT1886"
		"SMPTE_ST2084"
color_primaries	enum	
		Values
		"REC709"
		"REC2020"
		"DCI-P3-D65"
		"DCI-P3-Theater"
mastering_luminance	float	
content_aspect_ratio	enum	
		Values
		"5:4"
		"4:3"
		"16:10"
		"16:9"
		"1.85:1"
		"2.20:1"
		"2.35:1"
		"2.37:1"

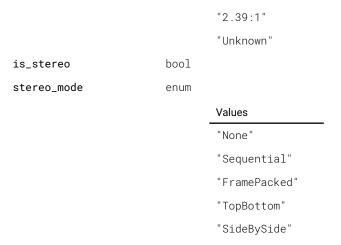


image.connector.l2displayportc.edid.selected

Selected EDID for connector



Access: RW

Name	Туре
selected	string

image.connector.l2displayportc.signalrange

Override the detected signal signal range. Set to Auto for automatic control.



Access: RW

Name	Type
signalrange	enum
	Values
	"Auto"
	"0-255"
	"16-235"

image.connector.l2displayportd.colorprimaries

Override the detected signal color primaries. Set to Auto for automatic control.



Name	Туре
colorprimaries	enum
	Values
	"Auto"
	"Uncorrected"
	"REC709"

"REC2020"

"DCI-P3-D65"

"DCI-P3-Theater"

image.connector.l2displayportd.colorprimariesavailable

true if Color Primaries is available



Access: R

Name	Туре
colorprimariesavailable	bool

image.connector.l2displayportd.colorspace

Override the detected signal color space. Set to Auto for automatic control.



Access: RW

Name	Туре
colorspace	enum
	Values
	"Auto"
	"RGB"
	"REC709"
	"REC601"
	"REC2020"

image. connector. I2 display port d. detected signal

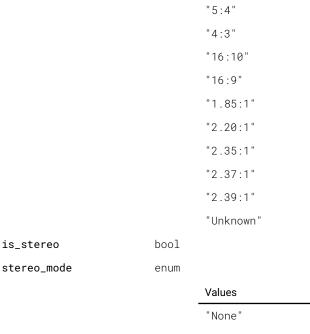
The signal information of the currently detected signal. If 'active' is false, there is no detected signal and the rest of the information should be disregarded. is_stereo indicates if stereo_mode is different from none.



Name	Туре	_
detectedsignal	{ }	
	Name	Туре
	active	bool
	name	string
	vertical_total	int
	horizontal_total	int
	vertical_resolution	int
	$horizontal_resolution$	int
	vertical_sync_width	int

		Reference Guide
vertical_front_porch	int	
vertical_back_porch	int	
horizontal_sync_width	int	
horizontal_front_porch	int	
horizontal_back_porch	int	
horizontal_frequency	float	
vertical_frequency	float	
pixel_rate	int	
scan	enum	
		Values
		"Progressive"
		"Interlaced"
bits_per_component	int	
color_space	enum	
		Values
		"RGB"
		"REC709"
		"REC601"
		"REC2020"
signal_range	enum	
		Values
		"0-255"
		"16-235"
chroma_sampling	enum	
		Values
		"4:4:4"
		"4:2:2"
		"4:2:0"
gamma_type	enum	
7,		Values
		"POWER"
		"sRGB"
		"REC_BT1886"
		"SMPTE_ST2084"
color_primaries	enum	
		Values
		"REC709"
		"REC2020"
		"DCI-P3-D65"
		"DCI-P3-Theater"
mastering_luminance	float	
content_aspect_ratio	enum	
		Values
		· uiuco

Reference Guide



"None"
"Sequential"
"FramePacked"
"TopBottom"
"SideBySide"

image.connector.l2displayportd.edid.selected

Selected EDID for connector



Access: RW

Name Type selected string

image.connector.l2displayportd.signalrange

Override the detected signal signal range. Set to Auto for automatic control.



Access: RW

Name	Туре
signalrange	enum
	Values
	"Auto"
	"0-255"
	"16-235"

image.contrast

Image contrast/gain. The value is normalized, 1 is default.



Access: RW

Name	Туре	
contrast	float	
	Constraints	
	Minimum	0
	Maximum	2
	Step size	1
	Precision	0.01

image.convergence.blue

Horizontal and vertical convergence offsets for blue: -2..2



Access: RW

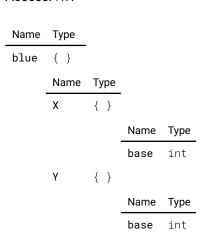


image.convergence.green

Horizontal and vertical convergence offsets for green: -2..2



Access: RW

image.convergence.red

Horizontal and vertical convergence offsets for red: -2..2



Access: RW

Name	Туре			
red	{ }			
	Name	Туре		
	Х	{ }		
			Name	Туре
			base	int
	Υ	{ }		
			Name	Туре
			base	int

image.display.desireddisplaymode

The desired display mode.



Access: RW

Name	Туре
desireddisplaymode	enum
	Values
	"Mono"
	"AutoStereo"
	"ActiveStereo"
	"NightVision"
	"IGPixelShift"

image.display.displaymode

The current display mode.



Access: R

Name	Туре
displaymode	enum
	Values
	"Mono"
	"AutoStereo"
	"ActiveStereo"
	"NightVision"
	"IGPixelShift"

image.display.frequency

The display frequency.



Access: R

Name	Type
frequency	float

image.display.synchronouslock

The display synchronous lock state.



Access: R

Name	Туре
synchronouslock	bool

image.gamma

Image gamma. Default is 2.2

MODELS All

Access: RW

Name	Туре	
gamma	float	
	Constraints	
	Minimum	1
	Maximum	3

image.intensity

Step size

Precision

1 0.1

Intensity

MODELS All

Access: RW

Name	Туре	
intensity	float	
	Constraints	
	Minimum	0
	Maximum	1
	Step size	0.1
	Precision	0.01

image.orientation

Description not provided



Access: RW

Name	Туре
orientation	enum
	Values
	"DESKTOP_FRONT"
	"DESKTOP_REAR"
	"CEILING_FRONT"
	"CEILING_REAR"

image.processing.blacklevel.basicblacklevel.bottom

Bottom edge.



Access: RW

Name	Type
bottom	int

image.processing.blacklevel.basicblacklevel.enable

Description not provided



Access: RW

Name	Type
enable	bool

image.processing.blacklevel.basicblacklevel.left

Left edge.



Access: RW

Name	Type
left	int

image.processing.blacklevel.basicblacklevel.level

Change the upper level of the black level adjustment



Name	Туре
level	int

Constraints	
Minimum	0
Maximum	65535
Step size	1
Precision	1

image.processing.blacklevel.basicblacklevel.right

Right edge.



Access: RW

Name	Type
right	int

image.processing.blacklevel.basicblacklevel.top

Top edge.



Access: RW

Name	Type
top	int

image.processing.blacklevel.bluegain

The gain blue for black level



Access: RW

Name	Туре	
bluegain	float	
	Constraints	
	Minimum	0
	Maximum	1
	Step size	1
	Precision	0.001

image.processing.blacklevel.file.enable

Enable/Disable black level correction



Name	Type
	hool

image.processing.blacklevel.file.selected

Currently selected file



Access: RW

Name	Туре
selected	strina

image.processing.blacklevel.greengain

The gain green for black level



Access: RW

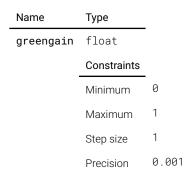


image.processing.blacklevel.redgain

The gain red for black level



Access: RW

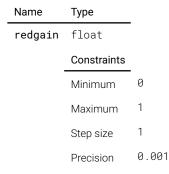


image.processing.blend.basicblend.bottom

Bottom blend edge.



Name	Туре
bottom	{ }

Name	Type
Start	int
Width	int

image.processing.blend.basicblend.enable

Description not provided



Access: RW

Name	Type
enable	bool

image.processing.blend.basicblend.left

Left blend edge.



Access: RW

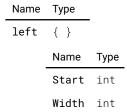


image.processing.blend.basicblend.right

Right blend edge.



Access: RW

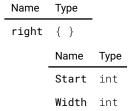
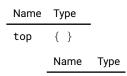


image.processing.blend.basicblend.top

Top blend edge.





Start int
Width int

image.processing.blend.file.enable

Enable/Disable file blend



Access: RW

Name Type enable bool

image.processing.blend.file.maxselected

Max number of selected files



Access: R

Name	Type
maxselected	int

image.processing.blend.file.selected

Currently selected files



Access: RW

Name	Туре
selected	[string]

image.processing.blend.scurve

S-Curve exponent strength.



Access: RW

Name	Туре	
scurve	float	
	Constraints	
	Minimum	1
	Maximum	4
	Step size	1
	Precision	0.1

image.processing.transportdelay.actual

Actual transport delay.



Access: R

Name	Type
20+112]	int

image.processing.transportdelay.desired

Desired transport delay.



Access: RW

Name	Туре
desired	int

image.processing.transportdelay.minimum

Minimum transport delay.



Access: R

Name	Type
minimum	int

image.processing.warp.bow.bottomleftu

U vector for bottom left corner. Positive angle is outwards.

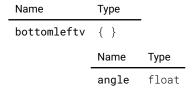
MODELS All

Access: RW

image.processing.warp.bow.bottomleftv

V vector for bottom left corner. Positive angle is outwards.





length float

image.processing.warp.bow.bottomrightu

U vector for bottom right corner. Positive angle is outwards.



Access: RW

Name	Туре	
bottomrightu	{ }	
	Name	Туре
	angle	float
	length	float

image.processing.warp.bow.bottomrightv

V vector for bottom right corner. Positive angle is outwards.



Access: RW

Name	Туре	
bottomrightv	{ }	
	Name	Туре
	angle	float
	length	float

image.processing.warp.bow.enable

Enable/Disable bow warp



Access: RW

Name	Type	
enable	bool	

image.processing.warp.bow.symmetric

Enable/Disable symmetric mode.



Access: RW

Name	Type	
symmetric	bool	

image.processing.warp.bow.topleftu

U vector for top left corner. Positive angle is outwards.

MODELS All

Access: RW

Name	Type	
topleftu	{ }	
	Name	Туре
	angle	float
	length	float

image.processing.warp.bow.topleftv

V vector for top left corner. Positive angle is outwards.



Access: RW

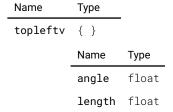


image.processing.warp.bow.toprightu

U vector for top right corner. Positive angle is outwards.



Access: RW

Name	Туре	
toprightu	{ }	
	Name	Туре
	angle	float
	length	float

image.processing.warp.bow.toprightv

V vector for top right corner. Positive angle is outwards.



Access: RW

Name	Type	
toprightv	{ }	
	Name	Туре
	angle	float
	length	float

image.processing.warp.enable

Enable/Disable all warp functions



Access: RW

Name	Type	
enable	bool	

image.processing.warp.file.enable

Enable/Disable file warp



Access: RW

Name	Type	
enable	bool	

image.processing.warp.file.selected

Currently selected file



Access: RW

Name	Туре	
selected	string	

image.processing.warp.fourcorners.bottomleft

Bottom left coordinate in output resolution. Negative values alowed to go outside displayed area.



Access: RW

name	туре	
bottomleft	{ }	
	Name	Туре
	х	int
	у	int

image.processing.warp.fourcorners.bottomright

Bottom right coordinate in output resolution. Negative values alowed to go outside displayed area.



Name	Туре	
bottomright	{ }	
	Name	Туре

X	int
v	int

image.processing.warp.fourcorners.enable

Enable/Disable FourCorners warp



Access: RW

Name	Type	
enable	bool	

image.processing.warp.fourcorners.screenheight

The height of the screen we are projecting on. Only used as in the ratio ScreenWidht/ScreenHeight, hence the unit is arbitrary.



Access: RW

Name	Туре
screenheight	float

image.processing.warp.fourcorners.screenwidth

The width of the screen we are projecting on. Only used as in the ratio ScreenWidht/ScreenHeight, hence the unit is arbitrary.

MODELS All

Access: RW

Name	Туре
screenwidth	float

image.processing.warp.fourcorners.topleft

Top left coordinate in output resolution. Negative values alowed to go outside displayed area.



Access: RW

Name	Type	
topleft	{ }	
	Name	Type
	x	int
	У	int

image.processing.warp.fourcorners.topright

Top right coordinate in output resolution. Negative values allowed to go outside displayed area.

MODELS All

Access: RW

Name	Type	
topright	{ }	
	Name	Туре
	Х	int
	у	int

image.resolution.alpha.size

The current resolution size (pixels x lines).

MODELS All

Access: R

Name	Туре	
size	{ }	
	Name	Туре
	pixels	int
	lines	int

image.resolution.beta.size

The current resolution size (pixels x lines).

MODELS All

Access: R

Name	Type	
size	{ }	
	Name	Туре
	pixels	int
	lines	int

image.resolution.display.size

The current resolution size (pixels x lines).

MODELS All

Access: R

INATTIC	туре	
size	{ }	
	Name	Туре
	pixels	int
	lines	int

image.resolution.osd.size

The current resolution size (pixels x lines).



Access: R

Name	Туре	
size	{ }	
	Name	Туре
	pixels	int
	lines	int

image.resolution.output.size

The current resolution size (pixels x lines).



Access: R

Name	Туре	
size	{ }	
	Name	Туре
	pixels	int
	lines	int

image.resolution.processing.size

The current resolution size (pixels x lines).



Access: R

Name	Туре	
size	{ }	
	Name	Туре
	pixels	int
	lines	int

image.resolution.resolution

The current resolution description.



Access: RW

Name	Туре
resolution	string

image.resolution.size

The current resolution size (pixels x lines).



Access: R

Name	Туре	
size	{ }	
	Name	Туре
	pixels	int
	lines	int

image.saturation

Image color saturation. The value is normalized, 1 is default.



Access: RW

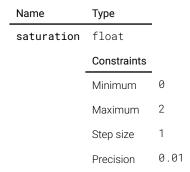


image.sharpness

Image sharpness. The value is normalized.



Access: RW

Name	Туре	
sharpness	int	
	Constraints	
	Minimum	-2
	Maximum	8
	Step size	1
	Precision	1

image.source.l1displayport.layout

Source layout



Name	Туре
layout	{ }

Name	Type
rows	int
columns	int
planes	int

image. source. I1hd baset 1. layout

Source layout



Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image. source. I1hd baset 2. layout

Source layout



Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l1hdmi.layout

Source layout



Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l1quadsdi.layout

Source layout



Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l1sdia.layout

Source layout



Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l1sdib.layout

Source layout



Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l1sdic.layout

Source layout



Name	Туре	
lavout	{ }	

Name	Type
rows	int
columns	int
planes	int

image.source.l1sdid.layout

Source layout



Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l2displayporta.layout

Source layout

MODELS UDX-4K32 UDX-4K22

Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l2displayportb.layout

Source layout

MODELS UDX-4K32 UDX-4K22

Access: R

Nume	турс	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l2displayportc.layout

Source layout



Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	nlanes	int

image.source.l2displayportd.layout

Source layout



Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l2dualdpab.layout

Source layout

MODELS UDX-4K32 UDX-4K22

Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l2dualdpac.layout

Source layout



Name	Туре
layout	{ }

Name	Type
rows	int
columns	int
planes	int

image.source.l2dualdpbd.layout

Source layout

MODELS UDX-4K32 UDX-4K22

Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l2dualdpcd.layout

Source layout

MODELS UDX-4K32 UDX-4K22

Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image. source. I2 dual headdpac. I a yout

Source layout

MODELS UDX-4K32 UDX-4K22

Access: R

Numc	турс	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l2dualheaddpbd.layout

Source layout



Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l2dualheaddualdpabcd.layout

Source layout



Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l2quadcolumndp.layout

Source layout

MODELS UDX-4K32 UDX-4K22

Access: R

Name	Туре	
layout	{ }	
	Name	Туре
	rows	int
	columns	int
	planes	int

image.source.l2quaddp.layout

Source layout



Name	Type
layout	{ }

Name	Type
rows	int
columns	int
planes	int

image.stereo.darktime

Darktime in us.



Access: RW

Name	Type
darktime	int

image.stereo.glassync.delay

Sync delay in us.



Access: RW

Name	Type
delay	int

image.stereo.glassync.delaymaximum

Maximum sync delay in us.



Access: R

Name	Туре
delaymaximum	int

image.stereo.glassync.delayminimum

Minimum sync delay in us.



Access: R

Name	Type
delayminimum	int

image.stereo.glassync.invert

Sync invert.



Name	Type
invert	bool

image.stereo.swapframepair

swap which stereo frames belong to each other



Access: RW

Name	Туре
swapframepair	bool

image.testpattern.selected

The unique id of the selected pattern



Access: RW

Name	Туре
selected	string

image.testpattern.show

Description not provided



Access: RW

Name	Type
show	bool

image.window.main.position

Window position



Access: R

name	туре	
position	{ }	
	Name	Туре
	Х	int
	у	int

image.window.main.scalingmode

The scaling mode to apply to the source



Access: RW

Name	Туре
scalingmode	enum
	Values
	"Fill"
	"OneToOne"
	"FillScreen"
	"Stretch"

image.window.main.size

Window size



Access: R

Name	Туре	
size	{ }	
	Name	Туре
	width	int
	height	int

image.window.main.source

The source that is dispayed in this window



Access: RW

Name	Туре
source	string

network.device.lan.carrier

Whether the device has carrier or not



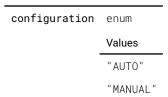
Access: R

Name	Type
carrier	bool

network.device.lan.configuration

The configuration method of the device: auto or manual





network.device.lan.devicetype

The general type of the network device



Access: R

Name	Туре
devicetype	enum
	Values
	"UNKNOWN"
	"WIRED"
	"WIRELESS"

network.device.lan.hwaddress

The active hardware (MAC) address



Access: R

Name	Туре
hwaddress	string

network.device.lan.ip4config

The current configuration for IP version 4



Access: R

Name	Туре	
ip4config	{ }	
	Name	Туре
	Address	string
	Mask	string
	Gateway	string
	NameServers	string

network.device.lan.ip4configmanual

Get/set the manual configuration for IP version 4



Access: RW

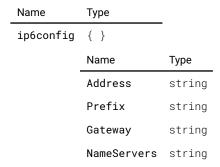
Name	Туре	
ip4configmanual	{ }	
	Name	Туре
	Address	string
	Mask	string
	Gateway	string
	NameServers	string

network.device.lan.ip6config

The current configuration for IP version 6



Access: R



network.device.lan.ip6configmanual

Get/set the manual configuration for IP version 4



Access: RW

name	туре	
ip6configmanual	{ }	
	Name	Туре
	Address	string
	Prefix	string
	Gateway	string
	NameServers	string

network.device.lan.speed

The speed of the device in Mbit/s



Name	Туре
speed	int

network.device.lan.state

The current state of the device



Access: R

Name	Туре
state	enum
	Values
	"CONNECTED"
	"DISCONNECTED"

network.device.lan.stateinfo

Additional information about the device state. Can be empty



Access: R

Name Type stateinfo string

network.hostname

The host name



Access: RW

Name	Type
hostname	string

network.version

The Networking Service version



Access: R

Name	Type	
version	string	

notification.count

The number of notifications received and dismissed



Name Type count int

optics.filteravailable

Description not provided



Access: R

Name	Type
filteravailable	hool

optics.focus.calibrationstate

Current calibration state



Access: R

Name	Туре
calibrationstate	enum
	Values
	"Unknown"
	"0k"
	"Busy"
	"Error"
	"NotImplemented"

optics.focus.enabled

Enabled state

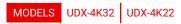


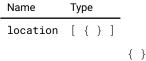
Access: RW

Name	Туре
enabled	bool

optics.focus.location

Saved locations



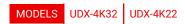


Name	Type
key	string

value int

optics.focus.maxposition

Maximum available position



Access: R

Name	Type
maxposition	int

optics.focus.minposition

Minimum available position



Access: R

Name	Type
minposition	int

optics.focus.position

Current position



Access: R

Name	Type
nosition	int

optics.focus.safetocalibrate

Safe to calibrate



Access: R

Name	Type
safetocalibrate	bool

optics.focus.safetooperate

Safe to operate state



Name	Type
safetooperate	bool

optics.focus.state

Current state

MODELS UDX-4K32 UDX-4K22

Access: R

Name	Туре
state	enum
	Values
	"Stopped"
	"Running"
	"Calibrating"
	"Homing"

optics.focus.target

Desired target

MODELS UDX-4K32 UDX-4K22

Access: RW

Name	Type
target	int

optics.lens

Description not provided

MODELS All

Name	Type	
lens	{ }	

Name	Type
ID	int
PowerID	int
Name	string
Description	string
Zoom	bool
ZoomForwardSpeed	float
ZoomReverseSpeed	float
ZoomPosition	bool
Focus	bool
FocusForwardSpeed	float
FocusReverseSpeed	float
FocusPosition	bool
Iris	bool

IrisForwardSpeed float
IrisReverseSpeed float
IrisResetTime float
IrisPosition bool
IrisResetPosition int

optics.lenspresent

Lens present



Access: R

Name	Type
lenspresent	hoo1

optics.lensshift.horizontal.calibrationstate

Current calibration state



Access: R

Name	Туре
calibrationstate	enum
	Values
	"Unknown"
	"0k"
	"Busy"
	"Error"
	"NotImplemented"

optics.lensshift.horizontal.enabled

Enabled state



Access: RW

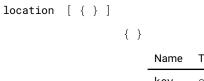
Name	Туре	
enabled	bool	

optics.lensshift.horizontal.location

Saved locations



Name	Туре



Name	Type
key	string
value	int

optics.lensshift.horizontal.maxposition

Maximum available position



Access: R

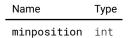
Name	Type
maxposition	int

optics.lensshift.horizontal.minposition

Minimum available position



Access: R



optics.lensshift.horizontal.position

Current position



Access: R

optics.lensshift.horizontal.safetocalibrate

Safe to calibrate



Access: R

Name	Туре
safetocalibrate	bool

optics.lensshift.horizontal.safetooperate

Safe to operate state



Access: R

Name	Type
cafatoonarata	hool

optics.lensshift.horizontal.state

Current state



Access: R

Name	Туре
state	enum
	Values
	"Stopped"
	"Running"
	"Calibrating"
	"Homing"

optics.lensshift.horizontal.target

Desired target



Access: RW

Name Type target int

optics.lensshift.vertical.calibrationstate

Current calibration state



Access: R

Name	Туре
calibrationstate	enum
	Values
	"Unknown"
	"0k"
	"Busy"
	"Error"
	"NotImplemented"

optics.lensshift.vertical.enabled

Enabled state



Access: RW

Name	Type
enahled	hool

optics.lensshift.vertical.location

Saved locations



Access: R

Name	Ty	/pe	!				
location	[{	}]			
					{ }		
						Name	Туре
						key	string
						value	int

optics.lensshift.vertical.maxposition

Maximum available position



Access: R

Name	Type
maxposition	int

optics.lensshift.vertical.minposition

Minimum available position



Access: R

Name	Type
minnosition	int

optics.lensshift.vertical.position

Current position



Access: R

Name	Туре
position	int

optics.lensshift.vertical.safetocalibrate

Safe to calibrate



Access: R

Name	Type
safetocalibrate	hool

optics. lens shift. vertical. safeto operate

Safe to operate state



Access: R

Name	Туре
safetooperate	bool

optics.lensshift.vertical.state

Current state



Access: R

Name	Туре
state	enum
	Values
	"Stopped"
	"Running"
	"Calibrating"
	"Homing"

optics.lensshift.vertical.target

Desired target



Access: RW

Name Type target int

optics.shutter.enabled

Enabled state of motor



Access: RW

Name Type

enabled bool

optics.shutter.position

Position of shutter



Access: R

Name	Туре
position	enum
	Values
	"Open"
	"Closed"

optics.shutter.target

Target of shutter



Access: RW

Name	Туре
target	enum
	Values
	"Open"
	"Closed"

optics.zoom.calibrationstate

Current calibration state



Access: R

Туре
enum
Values
"Unknown"
"0k"
"Busy"
"Error"
"NotImplemented"

optics.zoom.enabled

Enabled state

MODELS UDX-4K32 UDX-4K22

Access: RW

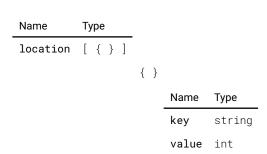
Name	Type
enabled	hool

optics.zoom.location

Saved locations



Access: R



optics.zoom.maxposition

Maximum available position



Access: R

Name	Туре
maxposition	int

optics.zoom.minposition

Minimum available position



Access: R

Name	Type
minposition	int

optics.zoom.position

Current position



Access: R

Name	Type
position	int

optics.zoom.safetocalibrate

Safe to calibrate



Access: R

Name	Type
safetocalibrate	bool

optics.zoom.safetooperate

Safe to operate state



Access: R

Name	Туре
safetooperate	bool

optics.zoom.state

Current state

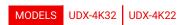
MODELS UDX-4K32 UDX-4K22

Access: R

Name	Туре
state	enum
	Values
	"Stopped"
	"Running"
	"Calibrating"
	"Homing"

optics.zoom.target

Desired target



Access: RW

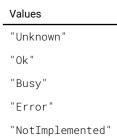
Name	Type
target	int

peripheral.frame.horizontal.calibrationstate

Current calibration state



Name	Туре
calibrationstate	enum



peripheral.frame.horizontal.state

Current state



Access: R

Name	Туре
state	enum
	Values
	"Stopped"
	"Running"
	"Calibrating"

peripheral.frame.rotation.calibrationstate

Current calibration state

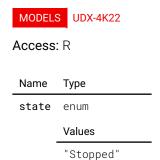


Access: R

Name	Туре
calibrationstate	enum
	Values
	"Unknown"
	"0k"
	"Busy"
	"Error"
	"NotImplemented"

peripheral.frame.rotation.state

Current state



```
"Running"
```

peripheral.frame.vertical.calibrationstate

Current calibration state



Access: R

Name	Туре
calibrationstate	enum
	Values
	"Unknown"
	"0k"
	"Busy"
	"Error"
	"NotImplemented"

peripheral.frame.vertical.state

Current state



Access: R

Name	Туре
state	enum
	Values
,	"Stopped"
	"Running"
	"Calibrating"

remotecontrol.address

The address of the remote control that the projector will respond to



Access: RW

Name	Туре	
address	int	
	Constraints	
	Minimum	1
	Maximum	31
	Step size	1
	Precision	1

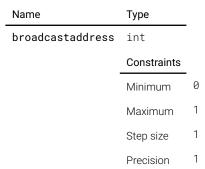
remotecontrol.broadcastaddress

[&]quot;Calibrating"

The broadcast address



Access: RW



remotecontrol.sensors.front.enable

Enable or disable the IR sensor



Access: RW

Name Type enable bool

remotecontrol.sensors.front.name

The display name of the IR sensor



Access: R

Name Type
name string

remotecontrol.sensors.rear.enable

Enable or disable the IR sensor



Access: RW

Name Type enable bool

remotecontrol.sensors.rear.name

The display name of the IR sensor



Access: R

Name Type

name string

remotecontrol.sensors.side.enable

Enable or disable the IR sensor



Access: RW

Name	Type
enable	hool

remotecontrol.sensors.side.name

The display name of the IR sensor



Access: R

Name	Туре
name	string

screen.hdrboost

The HDR intensity



Access: RW

Name	Туре	
hdrboost	float	
	Constraints	
	Minimum	0.8
	Maximum	1.2
	Step size	0.01
	Precision	0.1

screen.luminance

The maximum luminance measured on the screen in cd/m2



Name	Туре	
luminance	float	
	Constraints	
	Minimum	50
	Maximum	10000
	Step size	10

Precision

statistics.laserruntime.value

Counter value



Access: RW

Name Type value int

statistics.laserstrikes.value

Counter value



Access: RW

Name Type value int

statistics.projectorruntime.value

Counter value



Access: RW

Name Type value int

statistics.systemtime.value

Counter value



Access: RW

Name Type value int

statistics.uptime.value

Counter value



Access: RW

Name Type value int

system.articlenumber

Article number.



Access: R

Name	Туре
articlenumber	string

system.colorwheel

Article number of installed color wheel



Access: R

Name	Туре
colorwheel	string

system.colorwheelname

Name of installed color wheel



Access: R

Name	Туре
colorwheelname	string

system.eco.available

Returns true if state is available for this projector



Access: R

Name	Type
available	bool

system.eco.enable

Enable/disable the use of this state. Check if available first.



Access: RW

Name Type enable bool

system.error.timeout.duration

Time (in seconds) to wait in this state before entering lower state.



Access: RW

Name	Type
duration	int

system.error.timeout.enable

Enable/disable the timeout.



Access: RW

Name	Туре
enable	bool

system.familyname

Family name.



Access: R

Name	Туре
familyname	strina

system.firmwareversion

Firmware version.



Access: R

Name	Туре
firmwareversion	string

system.initialstate

State to transition to when the unit is started



Name	Туре
initialstate	enum
	Values
	"boot"
	"eco"
	"standby"

- "ready"
- "conditioning"
- "on"
- "service"
- "deconditioning"
- "error"

system.license.applicable

Applicability of the license file.



Access: R

Name Type applicable bool

system.license.available

Availability of a license file.



TODELO

Access: R

Name Type available bool

system.license.option.flexbrightness.enabled

State of the flex brightness option.

MODELS UDX-4K32

Access: R

Name Type enabled bool

system.license.option.flexbrightness.maximumlightoutput

The maximum light output.

MODELS UDX-4K32

Access: R

Name Type maximumlightoutput int

system. license. option. flex brightness. maximum light output attempts left and the control of the control o

Number of attempts left to set the maximum light output.

Access: R

Name	Туре
mavimumliahtoutnutattemntsleft	int

system.license.option.flexbrightness.maximumlightoutputs

List of valid maximum light outputs.



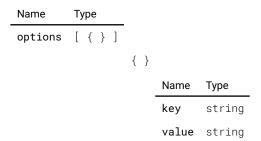
Name Type
maximumlightoutputs [int]

system.license.options

A dictionary of options and their values.



Access: R



system.license.valid

Validity of the license file.



Access: R

Name	Type
valid	bool

system.modelname

Model name.



Access: R

Name	Туре
modelname	string

system.on.timeout.duration

Time (in seconds) to wait in this state before entering lower state.

MODELS All

Access: RW

Name	Type
duration	int

system.on.timeout.enable

Enable/disable the timeout.

MODELS All

Access: RW

Name	Type
enable	bool

system.ready.timeout.duration

Time (in seconds) to wait in this state before entering lower state.

MODELS All

Access: RW

Name	Type
duration	int

system.ready.timeout.enable

Enable/disable the timeout.

MODELS All

Access: RW

Name	Type
enable	bool

system.serialnumber

Serial number.



Access: R

Name	Type
serialnumber	string

system.standby.available

Returns true if state is available for this projector



Name	Type
available	hoo1

system.standby.enable

Enable/disable the use of this state. Check if available first.



Access: RW

Name	Туре
enable	bool

system.standby.timeout.duration

Time (in seconds) to wait in this state before entering lower state.



Access: RW

Name	Type
duration	int

system. stand by. time out. enable

Enable/disable the timeout.



Access: RW

Name	Type
enable	bool

system.state

The current state of the unit



Name	Туре
state	enum
	Values
	"boot"
	"eco"
	"standby"
	"ready"
	"conditioning"
	"on"
	"service"

```
"deconditioning"
```

ui.access.enduser

True and available when the user has end user access privileges.



Access: R

Name	Type	
enduser	bool	

ui.backlight.state

Description not provided



Access: RW

Name	Type	
state	enum	
	Values	
	"Off"	
	"0n"	
	"Auto"	

ui.backlight.timeout

The amount of seconds after which the lcd backlight will be switched off when the menu and stealth mode are not active and there is no activity.

MODELS All

Access: RW

Name	Type
timeout	int

ui.hasstealthmode

Description not provided



Access: R

Name	Туре
hasstealthmode	bool

ui.keyboardshortcut

DEPRECATED: Primary and secondary shortcut states

[&]quot;error"



Access: RW

Name	Туре	
keyboardshortcut	{ }	

Name	Type
Primary	enum

Values
"NONE"
"INPUT"
"LENS"
"PATTERN"
"SHUTTER"
"INPUT_RC"
"LCD_RC"
"PATTERN_RC"
"RGB_RC"
"DEFAULT_RC"
"MACRO_RC"

Secondary enum

Values
"NONE"
"INPUT"
"LENS"
"PATTERN"
"SHUTTER"
"INPUT_RC"
"LCD_RC"
"PATTERN_RC"
"RGB_RC"
"DEFAULT_RC"
"MACRO_RC"

ui.layer.adjustment.enable

Enable or disable the layer. When enabled, the OSD will be hidden.



Access: RW

Name Type enable bool

ui.layer.adjustment.icon

The icon to show next to the value



Access: RW

Name	Type	
icon	string	

ui.layer.adjustment.layerposition

Placement of the layer related to the full screen.



Access: RW

Name	Туре	
layerposition	enum	
	Values	
	"TopLeft"	
	"Тор"	
	"TopRight"	
	"Right"	
	"BottomRight"	
	"Bottom"	
	"BottomLeft"	
	"Left"	
	"Center"	

ui.layer.adjustment.rangedvalue

Show a ranged value in the overlay



Access: RW

Name	Туре	
rangedvalue	{ }	
	Name	Туре
	Min	float
	Max	float
	Value	float
	Title	string

ui.layer.adjustment.showleftrightbuttons

Show/hide the left/right arrow buttons



Name	Type
showleftrightbuttons	bool

ui.layer.adjustment.showupdownbuttons

Show/hide the up/down arrow buttons



Access: RW

Name	Type
showupdownbuttons	bool

ui.layer.advancedblend.drawing

Drawing commands in the form of a JSON object



Access: RW

Name	Type	
drawing	string	

ui.layer.advancedblend.enable

Enable or disable the layer



Access: RW

Name	Type
enable	bool

ui.layer.advancedblend.palette

Color palette that can be used when drawing the blend layer



Access: RW

Name	Туре	
palette	[string]

ui.layer.basicblacklevel.color

The edge color, e.g '#ffff00' or 'rgba(255,255,0,0.5)



Name	Type	
color	string	

ui.layer.basicblacklevel.enable

Enable or disable the layer



Access: RW

Name	Type
enahle	hool

ui.layer.basicblacklevel.selection

Toggle edge selection



Access: RW

Name	Type	
selection	{ }	
	Name	Туре
	Bottom	bool
	Left	bool
	Right	bool
	Тор	bool

ui.layer.basicblacklevel.selectioncolor

The color to use for selected edges, e.g '#ff0000' or 'rgba(255,0,0,0.5)'



Access: RW

Name	Туре
selectioncolor	strina

ui.layer.basicblend.color

The edge color, e.g '#ffff00' or 'rgba(255,255,0,0.5)



Access: RW

Name	Type	
color	string	

ui.layer.basicblend.enable

Enable or disable the layer



Name	Type
enable	hool

ui.layer.basicblend.selection

Toggle edge selection



Access: RW

Name	Type	
selection	{ }	
	Name	Туре
	Bottom	bool
	Left	bool
	Right	bool
	Тор	bool

ui.layer.basicblend.selectioncolor

The color to use for selected edges, e.g '#ff0000' or 'rgba(255,0,0,0.5)'



Access: RW

Name	Type
selectioncolor	strina

ui.layer.fourcorner.cornercolor

The color to use for unselected corners, e.g '#ff0000' or 'rgba(255,0,0,0.75)'



Access: RW

Name	Type
cornercolor	string

ui.layer.fourcorner.enable

Enable or disable the layer



Access: RW

Name	Type
enable	bool

ui.layer.fourcorner.linecolor

The line color, e.g '#ffff00' org 'rgba(255,255,0,0.5)'

MODELS All

Access: RW

Name	Type
linecolor	string

ui.layer.fourcorner.lines

Show or hide lines between the corners



Access: RW

Name	Type
lines	bool

ui.layer.fourcorner.selection

Toggle corner selection



Access: RW

Name	Туре	
selection	{ }	
	Name	Туре
	TopLeft	bool
	TopRight	bool
	BottomRight	bool
	BottomLeft	bool

ui.layer.fourcorner.selectioncolor

The color to use for selected corners, e.g '#ff0000' or 'rgba(255,0,0,0.75)'



Access: RW

Name	Туре
selectioncolor	string

ui.layer.grid.color

Default color for grid points, e.g '#ff0000' or 'rgba(0,0,255,0.5)'



Name	Туре
color	string

ui.layer.grid.enable

Enable or disable the layer



Access: RW

Name	Type
enable	bool

ui.layer.grid.lines

DEPRECATED. Use ShowLines instead. This is for backwards compability

MODELS All

Access: RW

Name	Type
lines	bool

ui.layer.grid.mark

List of row,column and color triplets for marking points in the grid. The color is specified as '#ff00ff' or 'rgba(0,255,255,0.75)'

MODELS All

Access: RW

Name Type

		٠						
mark	[{	}]				
					{ }			
						Name	Туре	
						X	int	
						Υ	int	

Color string

ui.layer.grid.points

Number of grid points



Name	Туре
points	enum
	Values
	"2x2"
	"3x3"
	"5x5"

```
"9x9"
```

"17x17"

"33x33"

ui.layer.grid.showlines

Toggle drawing lines between grid points



Access: RW

Name	Type
showlines	hool

ui.layer.grid.showpoints

Toggle drawing grid points



Access: RW

Name	Туре
showpoints	bool

ui.menu

Show or hide the menu



Access: RW

Name	Type
menu	bool

ui.menuposition

Placement of menu related to full screen.



Name	Туре
menuposition	enum
	Values
	"TopLeft"
	"Тор"
	"TopRight"
	"Right"
	"BottomRight"
	"Bottom"

```
"BottomLeft"
```

"Left"

"Center"

ui.minimize

Minimize the menu when it is enabled



Access: RW

Name	Type
minimize	hoo1

ui.notificationfiltercodes

Filter display of notifications by notification code



Access: RW

Name	Туре	
notificationfiltercodes	[string]	

ui.notificationfilterseverity

Filter display of notifications by severity



Access: RW

Name	Туре
notificationfilterseverity	enum
	Values
	"CRITICAL"
	"ERROR"
	"WARNING"
	"INFO"
	"NONE"

ui.osd

Enable or disable on screen display



Access: RW

Name	Туре	
osd	bool	

ui.poweroffhint

When true, a dialog shows info about powering down



Access: RW

Name	Type
noweroffhint	hool

ui.sourcesignal

Show/hide the source signal information popup



Access: RW

Name	Type
sourcesignal	bool

ui.sourcesignalposition

Placement of the source signal information



Access: RW

Name	Туре
sourcesignalposition	enum
	Values
	"TopLeft"
	"Тор"
	"TopRight"
	"Right"
	"BottomRight"
	"Bottom"
	"BottomLeft"
	"Left"
	"Center"

ui.stealthmode

When the projector is in stealth mode, all controllable LEDs are switched off.



Name	Туре
stealthmode	enum
	Values
	 "Off"

"0n"

Methods

Alphabetical list of all methods

dmx.listchannels

Return a list of available channel names



This method does not require any parameters.

Return value

Name	Туре	
modes	[string]

dmx.listmodes

Return a list of all modes



This method does not require any parameters.

Return value

Name	Type	
modes	string	1

environment.getalarminfo

Description not provided



This method does not require any parameters.

Name	Туре	
alarminfo	[{ }]	
		{ }

Name	Туре
severity	string
timestamp	string
source	string
description	string
custommessage	string

environment.getcontrolblocks

Description not provided



Parameters

Name	Туре	
type	enum	
	Values	
	"Sensor"	
	"Filter"	
	"Controller"	
	"Actuator"	
	"Alarm"	
	"GenericBlock"	
valuetype	enum	
	Values	
	"Temperature"	
	"Speed"	
	"PWM"	
	"Voltage"	
	"Current"	
	"Power"	
	"Altitude"	
	"Pressure"	
	"Humidity"	
	"ADC"	
	"Coordinate"	
	"Peltier"	
	"Waveform"	
	"Average"	
	"Delay"	
	"Difference"	
	"Interpolation"	
	"Limit"	
	"Median"	
	"Noise"	
	"Weighting"	
	"Comparison"	
	"Threshold"	
	"Formula"	
	"Driver"	
	"PID"	

```
"Mode"

"State"

"Pump"

"Resistance"

"Simulation"

"Constant"

"Manual"

"Range"

"Any"
```

Return value

```
        Name
        Type

        blocks
        [ { } ]

        { }
        Name
        Type

        key
        string

        value
        float
```

illumination.clo.engage

Engage CLO at the current light level



This method does not require any parameters.

This method has no return value.

illumination.laser.getserialnumber

Description not provided



This method does not require any parameters.

Return value

Name	Туре	
value	string	

image.color.p7.custom.copypresettocustom

Description not provided



Parameters

Name Type

presetname string

This method has no return value.

image.color.p7.custom.resetpreset

Reset preset back to its default values



Parameters

Name	Туре	
presetname	strina	

This method has no return value.

image.color.p7.custom.resettonative

Description not provided



This method does not require any parameters.

This method has no return value.

image.color.rgbmode.nextrgbmode

Change to the next RGB mode. Lets you cycle through all possible modes.



This method does not require any parameters.

This method has no return value.

image.connector.l1displayport.edid.list

List system EDIDs available for this connector



This method does not require any parameters.

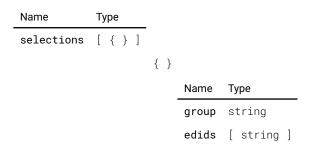


image.connector.l1hdbaset1.edid.list

List system EDIDs available for this connector

MODELS All

This method does not require any parameters.

Return value

```
Name Type
selections [ { } ]

{ }

Name Type
group string
edids [ string ]
```

image.connector.l1hdbaset2.edid.list

List system EDIDs available for this connector

MODELS All

This method does not require any parameters.

Return value

```
        Name
        Type

        selections
        [ { } ]

        { }
        Name
        Type

        group
        string

        edids
        [ string ]
```

image.connector.l1hdmi.edid.list

List system EDIDs available for this connector



This method does not require any parameters.

```
        Name
        Type

        selections
        [ { } ]

        { }
        Name
        Type

        group
        string

        edids
        [ string ]
```

image.connector.l2displayporta.edid.list

List system EDIDs available for this connector

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.

Return value

```
        Name
        Type

        selections
        [ { } ]

        { }
        Name
        Type

        group
        string

        edids
        [ string ]
```

image.connector.l2displayportb.edid.list

List system EDIDs available for this connector

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.

Return value

```
        Name
        Type

        selections
        [ { } ]

        { }
        Name
        Type

        group
        string

        edids
        [ string ]
```

image.connector.l2displayportc.edid.list

List system EDIDs available for this connector

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.

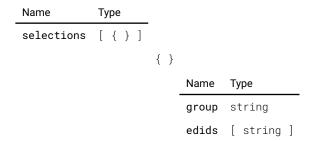


image.connector.l2displayportd.edid.list

List system EDIDs available for this connector

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.

Return value

```
Name Type
selections [ { } ]

{ }

Name Type
group string
edids [ string ]
```

image.connector.list

Description not provided



This method does not require any parameters.

Return value

Name	Туре
connectors	[string]

image.display.listdisplaymodes

List possible display modes.



This method does not require any parameters.

Return value

Name	Туре	
displaymodes	[enum]	
	Values	
	"Mono"	
	"AutoStereo"	
	"ActiveStereo"	
	"NightVision"	
	"IGPixelShift"	

image.processing.blacklevel.basicblacklevel.getblacklevelarea

Returns the four boxes describing the black level edges.



Parameters

Name	Туре
resolution_width	float
resolution_height	float

Return value

```
        Name
        Type

        output
        { }

        Name
        Type

        Top
        { }
```

Name Type
Start1 { }

Name Type
X float
Y float

 ${\tt Start2} \quad \{ \ \ \}$

Name Type
X float
Y float

 ${\tt Bottom} \quad \{\ \}$

Name Type
Start1 { }

Name Type
X float
Y float

 ${\tt Start2} \quad \{ \ \ \}$

Name Type

X float

Y float

Left { }

Name Type
Start1 { }

Name Type
X float
Y float

Start2 { }

Name Type
X float
Y float

 $\textbf{Right} \quad \{\ \}$

Name	Тур	e	
Start1	{ }	+	
		Name	Туре
		X	float
		Υ	float
Start2	{ }	+	
		Name	Туре
		Х	float
		Υ	float

image.processing.blacklevel.basicblacklevel.getwarpedblacklevelarea

Returns the four boxes describing the black level edges, after warp.



Parameters

Name	Type
resolution_width	float
resolution_height	float

Return value

Name	Type	_
output	{ }	
	Name	Type
	Тор	{ }

Name	Type
Start1	{ }

Name	Туре
Х	float
Υ	float

 $\texttt{Start2} \quad \{ \ \ \}$

Name	Type
Χ	float
Υ	float

Bottom { }

Name	Type
Х	float

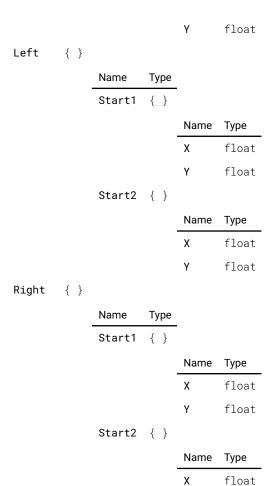


image.processing.blacklevel.file.delete

Υ

float

Deletes a file with the given name.



Parameters

Name	Type
filename	string

This method has no return value.

image.processing.blacklevel.file.list

Returns a list of available black level correction files



This method does not require any parameters.

Return value

Name	Туре
filenames	[string]

image.processing.blend.basicblend.getblendarea

Returns the four boxes describing the blend edges.



Parameters

Name	Type
resolution_width	float
resolution height	float

Return value

Name	Туре	_
output	{ }	_
	Name	Туре
	Тор	{ }

Name	Type
Start1	{ }

Name	Туре
Х	float
Υ	float

Start2 { }

Name	туре
Х	float
Υ	float

Width1 { }

 ${\tt Width2} \quad \{ \ \ \}$

Name	Type
Х	float
Υ	float

 ${\tt Bottom} \quad \{\ \}$

	Name	Туре
-	Χ	float
	Υ	float

Start2 { }

Name	Type
X	float
Υ	float

Width1 { }

Name Type

```
Χ
                                 float
                           Υ
                                 float
             Width2 { }
                           Name
                                Type
                                 float
                           Υ
                                 float
Left { }
             Name
                     Type
             Start1 { }
                           Name Type
                                 float
                           Χ
                           Υ
                                 float
             Start2 { }
                          Name Type
                           Χ
                                 float
                           Υ
                                 float
             Width1 { }
                           Name Type
                           Χ
                                 float
                           Υ
                                 float
             Width2 { }
                           Name Type
                           Χ
                                 float
                           Υ
                                 float
Right { }
             Name
             Start1 { }
                           Name Type
                           Χ
                                 float
                           Υ
                                 float
             Start2 { }
                          Name Type
                           Χ
                                 float
                           Υ
                                 float
             Width1 { }
                           Name
                                Type
                           Χ
                                 float
                           Υ
                                 float
             {\tt Width2} \quad \{ \ \ \}
                           Name Type
                           Χ
                                 float
```

Υ

float

image.processing.blend.basicblend.getwarpedblendarea

Returns the four boxes describing the blend edges, after warp.



Parameters

Name	Type
resolution_width	float
resolution height	float

Return value

name	туре	_
output	{ }	
	Name	Туре
	Тор	{ }

Name	Type		
Start1	{ }		
		Name	Туре
		Х	float
		Υ	float
Start2	{ }		
		Name	Туре
		Х	float
		Υ	float
Width1	{ }		
		Name	Туре
		Х	float
		Υ	float
Width2	{ }		
		Name	Type

 $\hbox{Bottom}\quad \{\ \}$

Name	Type	_	
Start1	{ }		
		Name	Туре
		Х	float
		Υ	float
Start2	{ }		

Χ

Υ

float

float

Name	Туре
Х	float
Υ	float

Width1 { }

Name	Type
Χ	float
Υ	float

Width2 { }

Name	Type
Х	float
٧	float

Left { }

Name	Type
Start1	{ }

Name	Туре
Х	float
Υ	float

Start2 { }

Name	Туре
Х	float
Υ	float

Width1 { }

Name	Туре
Χ	float
Υ	float

Width2 { }

Name	Туре
Х	float
Υ	float

Right { }

Name	Type		
Start1	{ }		

Name	Туре
Х	float
Υ	float

Start2 { }

Name	туре
Χ	float
Υ	float

Width1 { }

name	туре	
Χ	float	
Υ	float	

Width2 { }

Name	Type
------	------

Χ	float
Υ	float

image.processing.blend.file.delete

Deletes a file with the given name.



Parameters

Name	Type	
filename	strina	

This method has no return value.

image.processing.blend.file.list

Returns a list of available blend files



This method does not require any parameters.

Return value

Name	Туре		
filenames	[strina]		

image.processing.warp.file.delete

Deletes a file with the given name.



Parameters

Name	Type	
filename	strina	

This method has no return value.

image.processing.warp.file.list

Returns a list of available warp files



This method does not require any parameters.

Name	Туре	
filenames	[string]	

image.processing.warp.fourcorners.getscaledcorners

Get the corners scaled to the given resolution



Parameters

Ivallie	Type	
resolution	{ }	
	Name	Туре
	х	int
	у	int

Return value

Name	Туре			
corners	{ }			
	Name	Туре		
	TopLeft	{ }		
			Name	Туре
			х	int
			у	int
	TopRight	{ }		
			Name	Туре
			х	int
			у	int
	BottomLeft	{ }		
			Name	Туре
			x	int
			у	int
	BottomRight	{ }		
			Name	Туре
			x	int

image.processing.warp.warpscaledpoints

Takes an array of points and returns their warped equivalents.

int



Parameters

Name Type

X	floa
Υ	floa

resolution { }

Name	Type
Х	float
Υ	float

Return value

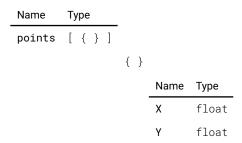


image.processing.warpgrid.getgrid

Get the current grid points as normalized and relative

MODELS All

This method does not require any parameters.

Return value

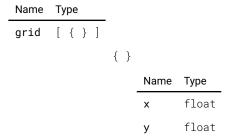


image.processing.warpgrid.getgridsize

Description not provided

MODELS All

This method does not require any parameters.

Return value

Name	Туре	
size	{ }	
	Name	Туре
	х	int
	у	int

image.processing.warpgrid.getscaledgrid

Get the current grid scaled to the given resolution



Parameters

Name	Type	
resolution	{ }	
	Name	Type
	х	int
	у	int

Return value

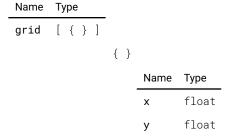


image.resolution.list

List possible resolutions.

MODELS All

This method does not require any parameters.

Return value

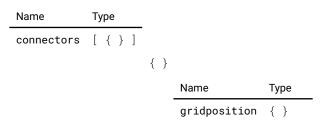
Name	Туре
resolutions	[string]

image.source.l1displayport.listconnectors

Get all connectors that are assigned to this source with their layout position

MODELS All

This method does not require any parameters.



Name	Type
row	int

```
column int plane int name string
```

image.source.l1hdbaset1.listconnectors

Get all connectors that are assigned to this source with their layout position



This method does not require any parameters.

Return value

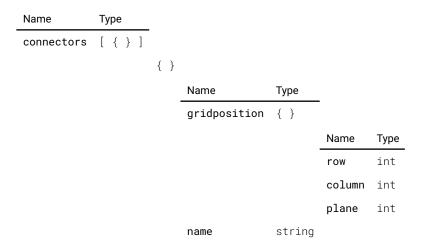


image.source.l1hdbaset2.listconnectors

Get all connectors that are assigned to this source with their layout position



This method does not require any parameters.

Return value

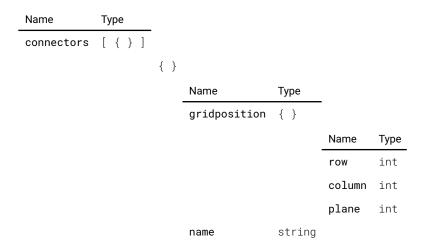


image.source.l1hdmi.listconnectors

Get all connectors that are assigned to this source with their layout position

MODELS All

This method does not require any parameters.

Return value

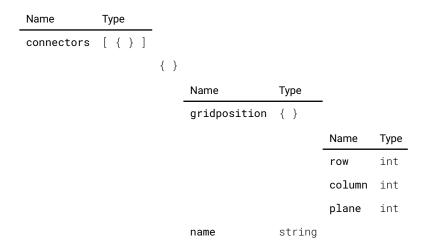


image.source.l1quadsdi.listconnectors

Get all connectors that are assigned to this source with their layout position

MODELS All

This method does not require any parameters.

Return value

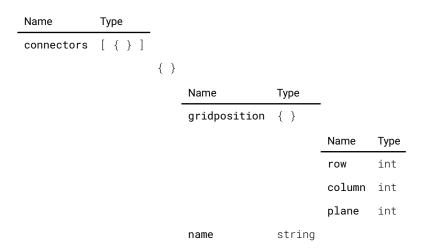


image.source.l1sdia.listconnectors

Get all connectors that are assigned to this source with their layout position



This method does not require any parameters.

Name	Type
connectors	[{ }]

image.source.l1sdib.listconnectors

Get all connectors that are assigned to this source with their layout position



This method does not require any parameters.

Return value

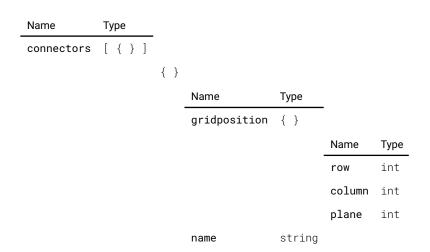


image.source.l1sdic.listconnectors

Get all connectors that are assigned to this source with their layout position



This method does not require any parameters.

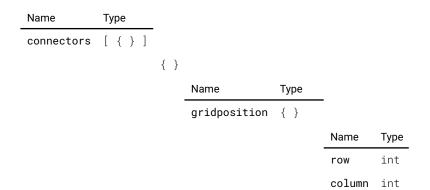


image.source.l1sdid.listconnectors

Get all connectors that are assigned to this source with their layout position



This method does not require any parameters.

Return value

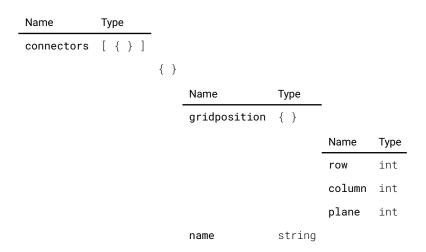
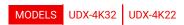


image.source.l2displayporta.listconnectors

Get all connectors that are assigned to this source with their layout position



This method does not require any parameters.

Return value

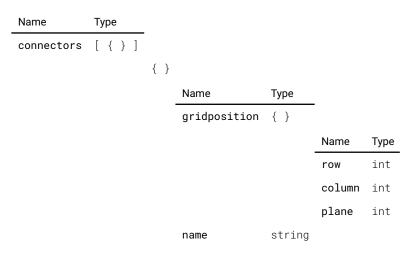


image.source.l2displayportb.listconnectors

Get all connectors that are assigned to this source with their layout position

This method does not require any parameters.

Return value

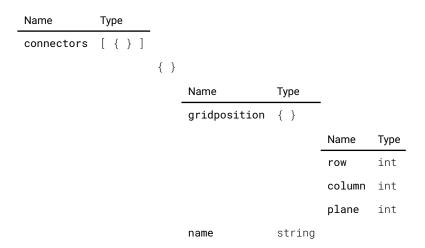


image.source.l2displayportc.listconnectors

Get all connectors that are assigned to this source with their layout position



This method does not require any parameters.

Return value

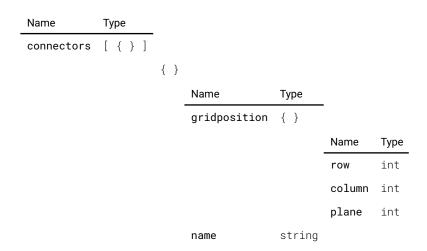
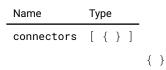


image.source.l2displayportd.listconnectors

Get all connectors that are assigned to this source with their layout position

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.



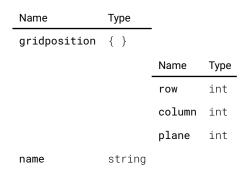


image.source.l2dualdpab.listconnectors

Get all connectors that are assigned to this source with their layout position

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.

Return value

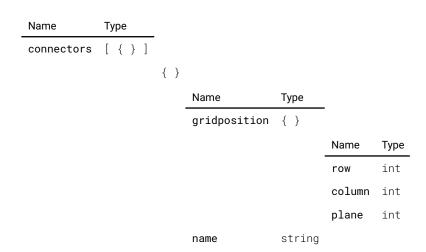
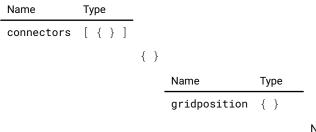


image.source.l2dualdpac.listconnectors

Get all connectors that are assigned to this source with their layout position

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.



Name	Type
row	int
column	int
plane	int

name

string

image.source.l2dualdpbd.listconnectors

Get all connectors that are assigned to this source with their layout position



This method does not require any parameters.

Return value

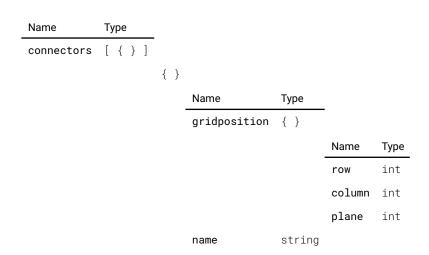


image.source.l2dualdpcd.listconnectors

Get all connectors that are assigned to this source with their layout position

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.

Return value

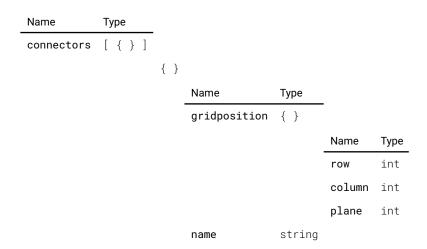


image.source.l2dualheaddpac.listconnectors

Get all connectors that are assigned to this source with their layout position



This method does not require any parameters.

Return value

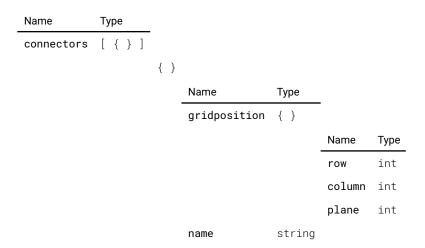
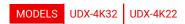


image.source.l2dualheaddpbd.listconnectors

Get all connectors that are assigned to this source with their layout position



This method does not require any parameters.

Return value

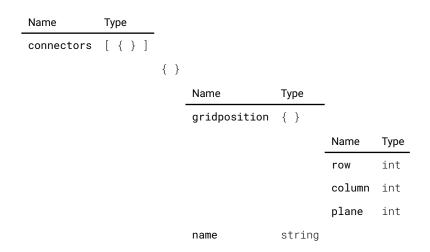
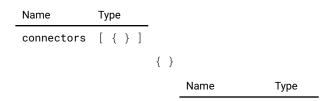


image. source. I2 dual head dual dpabcd. I is tconnectors

Get all connectors that are assigned to this source with their layout position

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.



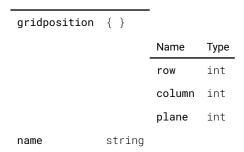


image.source.l2quadcolumndp.listconnectors

Get all connectors that are assigned to this source with their layout position

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.

Return value

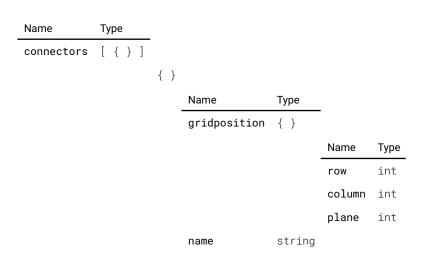


image.source.l2quaddp.listconnectors

Get all connectors that are assigned to this source with their layout position

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.

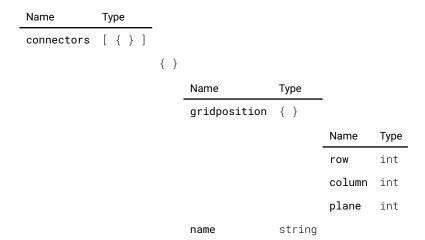


image.source.list

List all available sources



This method does not require any parameters.

Return value

Name	Туре	
sources	[string]

image.stereo.listdarktime

List all possible darktime values (in us).



This method does not require any parameters.

Return value

Name	Туре
darktime	[int]

image.testpattern.file.delete

Deletes a file with the given name.



Parameters

Name	Type	
filename	strina	

This method has no return value.

image.testpattern.file.list

Get a list of available custom uploaded patterns



This method does not require any parameters.

Return value

Name	Туре
patterns	[string]

image.testpattern.list

Get a list of available patterns

MODELS All

This method does not require any parameters.

Return value

```
NameTypepatterns[ { } ]
```

{ }

Name	Туре
name	string
location	string
id	string
above	bool
internal	bool
properties	[{ }]

{ }

Name	Type
key	string
value	string

image.testpattern.setproperties

Set the properties of a pattern

MODELS All

Parameters

Name	Туре			
id	string			
properties	[{ }]			
		{ }		
			Name	Туре
			key	string
			value	string

This method has no return value.

image.window.list

List all available windows



This method does not require any parameters.

Return value

Name Type

windows [string]

keydispatcher.sendclickevent

Send a key press event followed immediately by a key release event



Parameters

Name	Туре
key	enum
	Values
	"RC_SHUTTER_OPEN"
	"RC_SHUTTER_CLOSE"
	"RC_POWER_ON"
	"RC_POWER_OFF"
	"RC_OSD"
	"RC_LCD"
	"RC_PATTERN"
	"RC_RGB"
	"RC_ZOOM_PLUS"
	"RC_ZOOM_MINUS"
	"RC_SHIFT_LEFT"
	"RC_SHIFT_UP"
	"RC_SHIFT_RIGHT"
	"RC_SHIFT_DOWN"
	"RC_FOCUS_PLUS"
	"RC_FOCUS_MINUS"
	"RC_MENU"
	"RC_DEFAULT"
	"RC_BACK"
	"RC_UP"
	"RC_LEFT"
	"RC_OK"
	"RC_RIGHT"
	"RC_DOWN"
	"RC_ADDRESS"
	"RC_INPUT"
	"RC_MACRO"
	"RC_1"
	"RC_2"
	"RC_3"
	"RC_4"
	"RC_5"

"RC_7" "RC_8" "RC_9" "RC_0" "RC_ASTERISK" "RC_NUMBER" "KP_LEFT" "KP_UP" "KP_0K" "KP_RIGHT" "KP_DOWN" "KP_MENU" "KP_POWER"

"RC_6"

- "KP_BACK" "KP_OSD"
- "KP_LENS"
- "KP_PATTERN"
- "KP_SHUTTER"
- "KP_INPUT"
- "KP_STANDBY"

This method has no return value.

keydispatcher.sendpressevent

Send a key press event



Parameters

Name	Туре
key	enum
	Values
	"RC_SHUTTER_OPEN"
	"RC_SHUTTER_CLOSE"
	"RC_POWER_ON"
	"RC_POWER_OFF"
	"RC_OSD"
	"RC_LCD"
	"RC_PATTERN"
	"RC_RGB"
	"RC_ZOOM_PLUS"
	"RC_ZOOM_MINUS"
	"RC_SHIFT_LEFT"

- "RC_SHIFT_UP"
- "RC_SHIFT_RIGHT"
- "RC_SHIFT_DOWN"
- "RC_FOCUS_PLUS"
- "RC_FOCUS_MINUS"
- "RC_MENU"
- "RC_DEFAULT"
- "RC_BACK"
- "RC_UP"
- "RC_LEFT"
- "RC_OK"
- "RC_RIGHT"
- "RC_DOWN"
- "RC_ADDRESS"
- "RC_INPUT"
- "RC_MACRO"
- "RC_1"
- "RC_2"
- "RC_3"
- "RC_4"
- "RC_5"
- "RC_6"
- "RC_7"
- "RC_8"
- "RC_9"
- "RC_0"
- "RC_ASTERISK"
- "RC_NUMBER"
- "KP_LEFT"
- "KP_UP"
- "KP_0K"
- "KP_RIGHT"
- "KP_DOWN"
- "KP_MENU"
- "KP_POWER"
- "KP_BACK"
- "KP_OSD"
- "KP_LENS"
- "KP_PATTERN"
- "KP_SHUTTER"
- "KP_INPUT"
- "KP_STANDBY"

This method has no return value.

keydispatcher.sendreleaseevent

Send a key release event



Parameters

Name	Туре
key	enum
	Values
	"RC_SHUTTER_OPEN"
	"RC_SHUTTER_CLOSE"
	"RC_POWER_ON"
	"RC_POWER_OFF"
	"RC_OSD"
	"RC_LCD"
	"RC_PATTERN"
	"RC_RGB"
	"RC_ZOOM_PLUS"
	"RC_ZOOM_MINUS"
	"RC_SHIFT_LEFT"
	"RC_SHIFT_UP"
	"RC_SHIFT_RIGHT"
	"RC_SHIFT_DOWN"
	"RC_FOCUS_PLUS"
	"RC_FOCUS_MINUS"
	"RC_MENU"
	"RC_DEFAULT"
	"RC_BACK"
	"RC_UP"
	"RC_LEFT"
	"RC_OK"
	"RC_RIGHT"
	"RC_DOWN"
	"RC_ADDRESS"
	"RC_INPUT"
	"RC_MACRO"
	"RC_1"
	"RC_2"
	"RC_3"
	"RC_4"
	"RC_5"
	"RC_6"
	"RC_7"

"RC_8" "RC_9" "RC_0" "RC_ASTERISK" "RC_NUMBER" "KP_LEFT" "KP_UP" "KP_0K" "KP_RIGHT" "KP_DOWN" "KP_MENU" "KP_POWER" "KP_BACK" "KP_OSD" "KP_LENS" "KP_PATTERN" "KP_SHUTTER" "KP_INPUT" "KP_STANDBY"

This method has no return value.

led.activity

Activates the LEDS when enabled and restarts the LED timeout counter



This method does not require any parameters.

This method has no return value.

led.list

Description not provided



This method does not require any parameters.

Return value

Name Type

leds [string]

lightmeasurement.getlightoutput

Description not provided



This method does not require any parameters.

Return value

Name	Type	
lumens	int	

network.list

List of logical device id, e.g. 'wired1', 'wifi1'



This method does not require any parameters.

Return value

Name	Туре	
devices	[string]	

notification.dismiss

Dismiss the notification with the specified id



Parameters

Name	Туре
id	string
response	enum
	Values
	"NONE"
	"0K"
	"CANCEL"
	"IGNORE"
	"YES"
	"NO"
	"SUPPRESS"

This method has no return value.

notification.list

List all active notifications



This method does not require any parameters.



{ }

Name	Туре	_
severity	enum	
		Values
		"INFO"
		"CAUTION"
		"WARNING"
		"ERROR"
		"CRITICAL"
id	string	
code	string	
timestamp	string	
message	string	
timeout	string	
actions	[enum]	

Values
"NONE"
"OK"
"CANCEL"
"IGNORE"
"YES"
"NO"
"SUPPRESS"

notification.listsuppressed

Get a list of suppressed notification codes



This method does not require any parameters.

Return value

Name	Type	
list	[string]

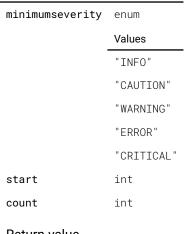
notification.log

List all saved notifications



Parameters

Name	Туре



Return value

Name	Туре
notifications	[{ }]

{ }

Name

severity	enum	
		Values
		"INFO"
		"CAUTION"
		"WARNING"
		"ERROR"
		"CRITICAL"
id	string	
code	string	
timestamp	string	
message	string	
timeout	string	
actions	[enum]	

Type

Values	
"NONE"	
"OK"	
"CANCEL"	
"IGNORE"	
"YES"	
"NO"	
"SUPPRESS"	

notification.suppress

Add a notification code to suppress (log but do not show on the LCD/OSD)



Parameters

Name	Type
code	string

This method has no return value.

notification.unsuppress

No longer suppress a certain notification code



Parameters

Name	Туре
code	strina

This method has no return value.

notification.unsuppressall

No longer suppress any notification codes



This method does not require any parameters.

This method has no return value.

optics.focus.addlocation

Add current position to location



Parameters

Name	Type
location	string

This method has no return value.

optics.focus.calibrate

Calibrate motor



This method does not require any parameters.

This method has no return value.

optics.focus.runforward

Run forward

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.

This method has no return value.

optics.focus.runforwardtime

Run forward for X milliseconds



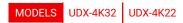
Parameters

Name	Type
milliseconds	int

This method has no return value.

optics.focus.runreverse

Run reverse



This method does not require any parameters.

This method has no return value.

optics.focus.runreversetime

Run reverse for X milliseconds



Parameters

Name	Type
milliseconds	int

This method has no return value.

optics.focus.setlocation

Set target to position at location



Parameters

Name	Type
location	string

This method has no return value.

optics.focus.stepforward

Step forward

MODELS UDX-4K32 UDX-4K22

Parameters

Name	Type
steps	int

This method has no return value.

optics.focus.stepreverse

Step reverse

MODELS UDX-4K32 UDX-4K22

Parameters

Name	Type
steps	int

This method has no return value.

optics.focus.stop

Stop

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.

This method has no return value.

optics.getvalidlensids

Description not provided

MODELS All

This method does not require any parameters.

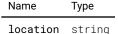
0 int1 string

optics.lensshift.horizontal.addlocation

Add current position to location



Parameters



This method has no return value.

optics.lensshift.horizontal.calibrate

Calibrate motor



This method does not require any parameters.

This method has no return value.

optics.lensshift.horizontal.runforward

Run forward



This method does not require any parameters.

This method has no return value.

optics.lensshift.horizontal.runforwardtime

Run forward for X milliseconds



Parameters



This method has no return value.

optics.lensshift.horizontal.runreverse

Run reverse



This method does not require any parameters.

This method has no return value.

optics.lensshift.horizontal.runreversetime

Run reverse for X milliseconds



Parameters

Name	Type
milliseconds	int

This method has no return value.

optics.lensshift.horizontal.setlocation

Set target to position at location



Parameters

Name	Type	
location	strina	

This method has no return value.

optics.lensshift.horizontal.stepforward

Step forward



Parameters

Name	Type
stens	int

This method has no return value.

optics.lensshift.horizontal.stepreverse

Step reverse



Parameters

Name	Type
steps	int

This method has no return value.

optics.lensshift.horizontal.stop

Stop



This method does not require any parameters.

This method has no return value.

optics.lensshift.vertical.addlocation

Add current position to location



Parameters

Name	Туре
location	strina

This method has no return value.

optics.lensshift.vertical.calibrate

Calibrate motor



This method does not require any parameters.

This method has no return value.

optics.lensshift.vertical.runforward

Run forward



This method does not require any parameters.

This method has no return value.

optics.lensshift.vertical.runforwardtime

Run forward for X milliseconds



Parameters

Name	Туре
milliseconds	int

This method has no return value.

optics.lensshift.vertical.runreverse

Run reverse



This method does not require any parameters.

This method has no return value.

optics.lensshift.vertical.runreversetime

Run reverse for X milliseconds



Parameters

Name	Type
milliseconds	int

This method has no return value.

optics.lensshift.vertical.setlocation

Set target to position at location



Parameters

Name	Туре
location	strina

This method has no return value.

optics.lensshift.vertical.stepforward

Step forward



Parameters



This method has no return value.

optics.lensshift.vertical.stepreverse

Step reverse



Parameters

Name	Type
etane	int

This method has no return value.

optics.lensshift.vertical.stop

Stop



This method does not require any parameters.

This method has no return value.

optics.setlensid

Description not provided



Parameters

Name	Type
lensid	int
powerlensid	int

This method has no return value.

optics.shifttocenter

Shift lens to center of allowed shift range



This method does not require any parameters.

This method has no return value.

optics.shutter.getobjectpath

Get object path of motor



This method does not require any parameters.

Name	Type
path	strina

optics.shutter.toggle

Toggle shutter position



This method does not require any parameters.

This method has no return value.

optics.zoom.addlocation

Add current position to location



Parameters

Name	Type
location	strina

This method has no return value.

optics.zoom.calibrate

Calibrate motor

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.

This method has no return value.

optics.zoom.runforward

Run forward

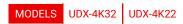
MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.

This method has no return value.

optics.zoom.runforwardtime

Run forward for X milliseconds



Parameters

Name	Type
milliseconds	int

This method has no return value.

optics.zoom.runreverse

Run reverse

MODELS UDX-4K32 UDX-4K22

This method does not require any parameters.

This method has no return value.

optics.zoom.runreversetime

Run reverse for X milliseconds

MODELS UDX-4K32 UDX-4K22

Parameters

Name Type milliseconds int

This method has no return value.

optics.zoom.setlocation

Set target to position at location

MODELS UDX-4K32 UDX-4K22

Parameters

Name Type
location string

This method has no return value.

optics.zoom.stepforward

Step forward

MODELS UDX-4K32 UDX-4K22

Parameters

Name Type steps int

This method has no return value.

optics.zoom.stepreverse

Step reverse

MODELS UDX-4K32 UDX-4K22

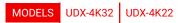
Parameters

Name	Type
ctanc	int

This method has no return value.

optics.zoom.stop

Stop



This method does not require any parameters.

This method has no return value.

peripheral.frame.horizontal.calibrate

Calibrate motor

MODELS UDX-4K22

This method does not require any parameters.

This method has no return value.

peripheral.frame.horizontal.runforward

Run forward

MODELS UDX-4K22

This method does not require any parameters.

This method has no return value.

peripheral.frame.horizontal.runreverse

Run reverse

MODELS UDX-4K22

This method does not require any parameters.

This method has no return value.

peripheral.frame.horizontal.stepforward

Step forward

MODELS UDX-4K22

Parameters

Name	Type
stens	int

This method has no return value.

peripheral.frame.horizontal.stepreverse

Step reverse



Parameters

Name	Type
steps	int

This method has no return value.

peripheral.frame.horizontal.stop

Stop

MODELS UDX-4K22

This method does not require any parameters.

This method has no return value.

peripheral.frame.rotation.calibrate

Calibrate motor

MODELS UDX-4K22

This method does not require any parameters.

This method has no return value.

peripheral.frame.rotation.runforward

Run forward

MODELS UDX-4K22

This method does not require any parameters.

This method has no return value.

peripheral.frame.rotation.runreverse

Run reverse



This method has no return value.

peripheral.frame.rotation.stepforward

Step forward



Parameters

Name	Type
stens	int

This method has no return value.

peripheral.frame.rotation.stepreverse

Step reverse



Parameters

Name	Type
steps	int

This method has no return value.

peripheral.frame.rotation.stop

Stop

MODELS UDX-4K22

This method does not require any parameters.

This method has no return value.

peripheral.frame.vertical.calibrate

Calibrate motor

MODELS UDX-4K22

This method does not require any parameters.

This method has no return value.

peripheral.frame.vertical.runforward

Run forward



This method has no return value.

peripheral.frame.vertical.runreverse

Run reverse

MODELS UDX-4K22

This method does not require any parameters.

This method has no return value.

peripheral.frame.vertical.stepforward

Step forward

MODELS UDX-4K22

Parameters

Name	Type
steps	int

This method has no return value.

peripheral.frame.vertical.stepreverse

Step reverse

MODELS UDX-4K22

Parameters

Name	Type
steps	int

This method has no return value.

peripheral.frame.vertical.stop

Stop

MODELS UDX-4K22

This method does not require any parameters.

This method has no return value.

remotecontrol.listsensors

Return a list of all the object names of the IR sensors



Return value

Name	Туре	
sensors	[string]	

statistics.laserruntime.getname

Name of the counter



This method does not require any parameters.

Return value

Name	Type
countername	strina

statistics.laserruntime.getunit

Unit of measurements



This method does not require any parameters.

Return value

Name	Туре
unit	enum
	Values
	"none"
	"hours"
	"minutes"
	"seconds"
	"number"
	"percent"

statistics.laserstrikes.getname

Name of the counter



This method does not require any parameters.

Return value

Name	Туре
countername	string

statistics.laserstrikes.getunit

Unit of measurements



This method does not require any parameters.

Return value

Name	Туре
unit	enum
	Values
	"none"
	"hours"
	"minutes"
	"seconds"
	"number"
	"percent"

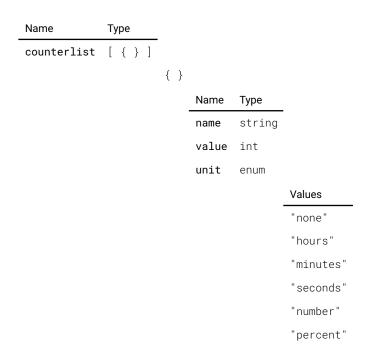
statistics.listcounters

List all counter names



This method does not require any parameters.

Return value



statistics.projectorruntime.getname

Name of the counter



Return value

Name	Туре
countername	strina

statistics.projectorruntime.getunit

Unit of measurements



This method does not require any parameters.

Return value

Name	Туре
unit	enum
	Values
	"none"
	"hours"
	"minutes"
	"seconds"
	"number"
	"percent"

statistics.systemtime.getname

Name of the counter



This method does not require any parameters.

Return value

Name	Type
countername	strina

statistics.systemtime.getunit

Unit of measurements



This method does not require any parameters.

Return value

Name	Туре
unit	enum
	Values

- "none"
- "hours"
- "minutes"
- "seconds"
- "number"
- "percent"

statistics.uptime.getname

Name of the counter



This method does not require any parameters.

Return value

Name	Туре
countername	string

statistics.uptime.getunit

Unit of measurements



This method does not require any parameters.

Return value

Name	Туре
unit	enum
	Values
	"none"
	"hours"
	"minutes"
	"seconds"
	"number"
	"percent"

system.activity

Signal user activity (resets timeout countdown timers)



This method does not require any parameters.

This method has no return value.

system.boards.getboardinfo

Get board properties for the specified board



Parameters

Name	Туре
boardname	strina

Return value

system.boards.getboardlist

Description not provided

MODELS All

This method does not require any parameters.

Return value

Name	Туре
boards	[string]

system.boards.getdeviceinfo

DEPRECATED. Use GetBoardInfo instead



Parameters

Name	Туре
boardname	string

Return value

system.boards.getmissingboardlist

Description not provided



This method does not require any parameters.

Return value

Name	Туре
hoards	[string]

system.boards.getmoduleinfo

Description not provided



Parameters

Name	Туре
boardname	string

Return value

system.getidentification

Description not provided



____<u>-</u>

Parameters

Name	Type
identification	string

Return value

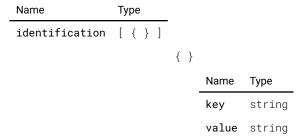
Name	Туре
value	strina

system.getidentifications

Description not provided



Return value



system.getsystemdate

Returns the system date as UTC time.



This method does not require any parameters.

Return value

Name	Туре	
date	{ }	
	Name	Туре
	year	int
	month	int
	day	int
	hour	int
	minute	int
	second	int

system.gotoeco

Set the device in the eco state



This method does not require any parameters.

This method has no return value.

system.gotoready

Set the device in the ready state



This method does not require any parameters.

This method has no return value.

system.license.option.flexbrightness.getmaximumlightoutputcode

226 of 253

Description not provided



Parameters

Name	Type	
lightoutput	int	
signature	string	

Return value

Name	Туре
code	string

system.license.option.flexbrightness.setmaximumlightoutput

Description not provided



Parameters

Name	Type	
code	string	
lightoutput	int	

This method has no return value.

system. license. option. flex brightness. set maximum light output code

Description not provided



Parameters

Name	Type	
lightoutput	int	
signature	string	
code	string	

This method has no return value.

system.listresetdomains

Returns the list of available reset domains



This method does not require any parameters.

Return value



system.poweroff

Power off the unit



This method does not require any parameters.

This method has no return value.

system.poweron

Power on the unit



This method does not require any parameters.

This method has no return value.

system.reset

Asynchronously starts reset of selected domains. The completion of the domains are signalled by one ore more 'Performed'-signals. Subsequent calls to 'ResetAll' or 'Reset' will fail until all domains have completed.



Parameters

Name	Туре	
domains	[enum]	
	Values	
	"ImageConnector"	
	"ImageSource"	
	"ImageFeatures"	
	"ImageRealColor"	
	"ImageWarp"	
	"ImageBlend"	
	"ImageOrientation"	
	"ImageResolution"	
	"ImageStereo"	
	"ImageDisplay"	
	"ImageTestPattern"	
	"ImageConvergence"	
	"UserInterface"	
	"Optics"	
	"Illumination"	
	"Network"	
	"Screen"	
	"System"	
	"LightMeasurement"	
	"Dmx"	

This method has no return value.

system.resetall

Asynchronously starts reset of all domains. The completion of the domains are signalled by one ore more 'Performed'signals. Subsequent calls to 'ResetAll' or 'Reset' will fail until all domains have completed.

MODELS All

This method does not require any parameters.

Return value

Name	Туре	
domains	[enum]	
	Values	
	"ImageConnector"	
	"ImageSource"	
	"ImageFeatures"	

- "ImageRealColor"
- "ImageWarp"
- "ImageBlend"
- "ImageOrientation"
- "ImageResolution"
- "ImageStereo"
- "ImageDisplay"
- "ImageTestPattern"
- "ImageConvergence"
- "UserInterface"
- "Optics"
- "Illumination"
- "Network"
- "Screen"
- "System"
- "LightMeasurement"
- "Dmx"

ui.settings.get

Get the value of the specified key



Parameters

Name	Type
key	string

Return value

Name	Type
value	string

ui.settings.getfonticons

Return a dictionary of icons for the specified category. Then icon is returned as a string containing the font set class name and the icon class name.





Parameters

Name	Туре	
category	enum	
	Values	
	"Source"	
	"Connector"	
	"TestPattern"	

Return value

Name	Туре			
dictionary	[{ }]			
		{ }		
			Name	Туре
			key	string
			value	string

ui.settings.geticons

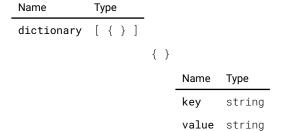
Return a dictionary of icons for the specified category. The icon is return as a SVG sprite name.



Parameters

Name	Туре	
category	enum	
	Values	
	"Source"	
	"Connector"	
	"TestPattern"	

Return value



ui.settings.keys

Return a list of all the keys



This method does not require any parameters.

Return value

Name	Туре
kevs	[string]

ui.settings.list

Return a list of key/value pairs of all the settings



This method does not require any parameters.

Return value

ui.settings.remove

Remove the specfied key and value



Parameters

Name	Туре
key	string

This method has no return value.

ui.settings.set

Set the key to the specified value



Parameters

Name	Туре
key	string
value	string

This method has no return value.

ui.togglestealthmode

This method is depreciated.



This method does not require any parameters.

This method has no return value.

Signals

Alphabetical list of all signals

modelupdated

Signals whenever functionality for an object appears or disappears

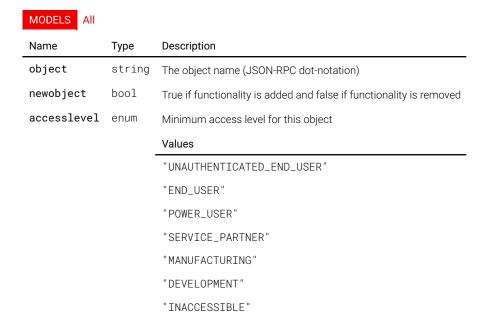


image.connector.l1displayport.edid.listchanged

Will be raised when list of EDIDs changed



This signal does contain any arguments.

image.connector.l1hdbaset1.edid.listchanged

Will be raised when list of EDIDs changed



This signal does contain any arguments.

image.connector.l1hdbaset2.edid.listchanged

Will be raised when list of EDIDs changed



This signal does contain any arguments.

image.connector.l1hdmi.edid.listchanged

Will be raised when list of EDIDs changed

MODELS All

This signal does contain any arguments.

image.connector.l2displayporta.edid.listchanged

Will be raised when list of EDIDs changed

MODELS UDX-4K32 UDX-4K22

This signal does contain any arguments.

image.connector.l2displayportb.edid.listchanged

Will be raised when list of EDIDs changed

MODELS UDX-4K32 UDX-4K22

This signal does contain any arguments.

image.connector.l2displayportc.edid.listchanged

Will be raised when list of EDIDs changed

MODELS UDX-4K32 UDX-4K22

This signal does contain any arguments.

image.connector.l2displayportd.edid.listchanged

Will be raised when list of EDIDs changed

MODELS UDX-4K32 UDX-4K22

This signal does contain any arguments.

image.processing.blacklevel.file.listchanged

This signal is fired every time the return value of List() changes

MODELS All

This signal does contain any arguments.

image.processing.blend.file.listchanged

This signal is fired every time the return value of List() changes

MODELS All

This signal does contain any arguments.

image.processing.warp.file.listchanged

This signal is fired every time the return value of List() changes

MODELS All

This signal does contain any arguments.

image.processing.warpgrid.changed

Fired when the grid changes, without the grid data as payload



This signal does contain any arguments.

image.processing.warpgrid.gridchanged

Description not provided

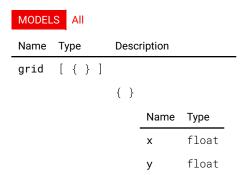


image.testpattern.added

Description not provided

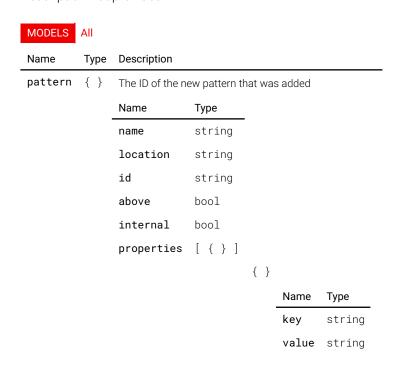


image.testpattern.changed

Description not provided



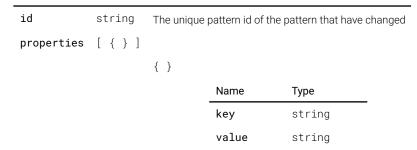


image.testpattern.file.listchanged

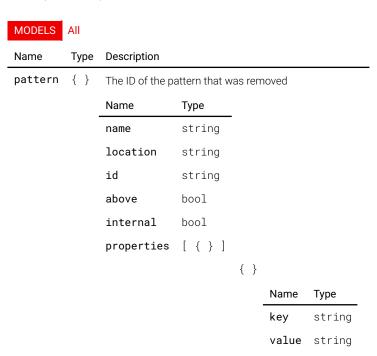
This signal is fired every time the return value of List() changes



This signal does contain any arguments.

image.testpattern.removed

Description not provided



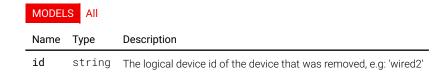
network.added

Raised when a new device has been added



network.removed

Raised when a device has been removed



notification.dismissed

Description not provided



Name	Туре	Description
id	string	The ID of the dismissed notification
response	enum	The reason or response for dismissing the notification
		Values
		"NONE"
		"OK"
		"CANCEL"
		"IGNORE"
		"YES"
		"NO"
		"SUPPRESS"

notification.emitted

Description not provided

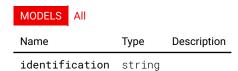


Name	Туре	Description		
notification	{ }	The new notif	cation	
		Name	Туре	
		severity	enum	•
				Values
				"INFO"
				"CAUTION"
				"WARNING"
				"ERROR"
				"CRITICAL"
		id	string	
		code	string	
		timestamp	string	
		message	string	
		timeout	string	
		actions	[enum]	
				Values
				"NONE"

"SUPPRESS"

system.identificationchanged

Will be raised whenever an identification is changed



system.license.licensechanged

Description not provided



This signal does contain any arguments.

system.performed

Emitted when one or more domains have completed resetting. This signal might be emitted several times, every time with one or more domains, until all requested domains are done resetting. 'All domains' in this context is either the list of domains supplied to a call to 'Reset' or the list of domains returned from 'ResetAll'.



ui.settings.added

Fired when a new key/value pair was added



Name	Type	Description
key	string	The key name
value	string	The kev value

ui.settings.changed

Fired when a key has an updated value



Name	Туре	Description
key	string	The key name
value	string	The value of the key

ui.settings.removed

Fired when a key was removed



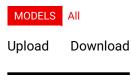
Name	Type	Description
key	string	The key name

Files

Alphabetical list of all file end points.

image.connector.edid.transfer

Upload and download EDID files



Yes

Example of file upload using the curl program.



Yes

pulse:~\$ curl -F file=@edid.dat http://192.168.1.100/api/image/connector/edid/transfer

Example of file download using the curl program on Linux and Mac OS X. Using the -0 and -J option makes curl use the file name suggestion of the server and will overwrite a file if it already exists.



pulse:~\$ curl -0 -J http://192.168.1.100/api/image/connector/edid/transfer

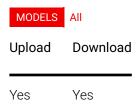
Example of file download using the PowerShell on Windows.



PS C:\Users\pulse> Invoke-WebRequest -Uri http://192.168.1.100/api/image/connector/edid/transfer - Method Get -OutFile edid.dat

image.processing.blacklevel.file.transfer

Upload and download black level correction file



Example of file upload using the curl program.



pulse:~\$ curl -F file=@blacklevel.dat http://192.168.1.100/api/image/processing/blacklevel/file/ transfer

Example of file download using the curl program on Linux and Mac OS X. Using the -0 and -J option makes curl use the file name suggestion of the server and will overwrite a file if it already exists.





 $pulse: \sim \$ \ curl \ -0 \ -J \ http://192.168.1.100/api/image/processing/blacklevel/file/transfer$

Example of file download using the PowerShell on Windows.



PS C:\Users\pulse> Invoke-WebRequest -Uri http://192.168.1.100/api/image/processing/blacklevel/file/ transfer -Method Get -OutFile blacklevel.dat

image.processing.blend.file.transfer

Upload and download blend files



Upload Download

Yes

Yes

Example of file upload using the curl program.





 $pulse: \sim \$ \ curl \ -F \ file=@blend.dat \ http://192.168.1.100/api/image/processing/blend/file/transfered by the control of the control of$

Example of file download using the curl program on Linux and Mac OS X. Using the -0 and -J option makes curl use the file name suggestion of the server and will overwrite a file if it already exists.





pulse:~\$ curl -0 -J http://192.168.1.100/api/image/processing/blend/file/transfer

Example of file download using the PowerShell on Windows.



PS C:\Users\pulse> Invoke-WebRequest -Uri http://192.168.1.100/api/image/processing/blend/file/transfer -Method Get -OutFile blend.dat

image.processing.warp.file.transfer

Upload and download warp files





Upload

Download

Yes

Yes

Example of file upload using the curl program.





pulse:~\$ curl -F file=@warp.dat http://192.168.1.100/api/image/processing/warp/file/transfer

Example of file download using the curl program on Linux and Mac OS X. Using the -0 and -J option makes curl use the file name suggestion of the server and will overwrite a file if it already exists.





pulse:~\$ curl -0 -J http://192.168.1.100/api/image/processing/warp/file/transfer

Example of file download using the PowerShell on Windows.



PS C:\Users\pulse> Invoke-WebRequest -Uri http://192.168.1.100/api/image/processing/warp/file/transfer -Method Get -OutFile warp.dat

image.testpattern.file.transfer

Upload and download test pattern images



Upload Download

Yes Yes

Example of file upload using the curl program.





pulse:~\$ curl -F file=@testpattern.dat http://192.168.1.100/api/image/testpattern/file/transfer

Example of file download using the curl program on Linux and Mac OS X. Using the -0 and -J option makes curl use the file name suggestion of the server and will overwrite a file if it already exists.





pulse:~\$ curl -0 -J http://192.168.1.100/api/image/testpattern/file/transfer

Example of file download using the PowerShell on Windows.



PS C:\Users\pulse> Invoke-WebRequest -Uri http://192.168.1.100/api/image/testpattern/file/transfer -Method Get -OutFile testpattern.dat

notification.logger.transfer

Download notification log files





Upload

Download

No

Yes

Example of file download using the curl program on Linux and Mac OS X. Using the -0 and -J option makes curl use the file name suggestion of the server and will overwrite a file if it already exists.





pulse:~\$ curl -0 -J http://192.168.1.100/api/notification/logger/transfer

Example of file download using the PowerShell on Windows.



PS C:\Users\pulse> Invoke-WebRequest -Uri http://192.168.1.100/api/notification/logger/transfer -Method Get -OutFile logger.dat

Property index

dmx.artnet	42	dmx.artnetnet	42
dmx.artnetuniverse	42	dmx.mode	42
dmx.monitor.channel01.function	42	dmx.monitor.channel01.offset	43
dmx.monitor.channel01.value	43	dmx.monitor.channel02.function	43
dmx.monitor.channel02.offset	43	dmx.monitor.channel02.value	43
dmx.monitor.channel03.function	44	dmx.monitor.channel03.offset	44
dmx.monitor.channel03.value	44	dmx.monitor.channel04.function	44
dmx.monitor.channel04.offset	44	dmx.monitor.channel04.value	44
dmx.monitor.channel05.function	45	dmx.monitor.channel05.offset	45
dmx.monitor.channel05.value	45	dmx.monitor.channel06.function	45
dmx.monitor.channel06.offset	45	dmx.monitor.channel06.value	46
dmx.monitor.channel07.function	46	dmx.monitor.channel07.offset	46
dmx.monitor.channel07.value	46	dmx.monitor.channel08.function	46
dmx.monitor.channel08.offset	47	dmx.monitor.channel08.value	47
dmx.monitor.channel09.function	47	dmx.monitor.channel09.offset	47
dmx.monitor.channel09.value	47	dmx.monitor.channel10.function	48
dmx.monitor.channel10.offset	48	dmx.monitor.channel10.value	48
dmx.monitor.channel11.function	48	dmx.monitor.channel11.offset	48
dmx.monitor.channel11.value	48	dmx.monitor.channel12.function	49
dmx.monitor.channel12.offset	49	dmx.monitor.channel12.value	49
dmx.monitor.channel13.function	49	dmx.monitor.channel13.offset	49
dmx.monitor.channel13.value	50	dmx.monitor.channel14.function	50
dmx.monitor.channel14.offset	50	dmx.monitor.channel14.value	50
dmx.monitor.connectionstate.active	50	dmx.shutdown	51
dmx.shutdowntimeout	51	dmx.startchannel	51
environment.alarmstate	51	firmware.firmwareversion	51
gsm.available	52	gsm.pin	52
gsm.pinstate	52	illumination.clo.availability	52
illumination.clo.enable	53	illumination.clo.scale	53
illumination.clo.setpoint	53	illumination.clo.state	53
illumination.sources.laser.ispowerlimited	54	illumination.sources.laser.maxpower	54
illumination.sources.laser.minpower	54	illumination.sources.laser.power	54
illumination.sources.laser.powerlimitreason	54	illumination.state	55
image.blackcontentdetection.dimminginterval	55	image.blackcontentdetection.enable	55
image.blackcontentdetection.sampleinterval	55	image.blackcontentdetection.state	56
image.blackcontentdetection.threshold	56	image.brightness	56
image.color.p7.custom.bluegain	56	image.color.p7.custom.bluelum	57
image.color.p7.custom.bluex	57	image.color.p7.custom.bluey	57
image.color.p7.custom.cmyreadable	57	image.color.p7.custom.cmywritable	57
image.color.p7.custom.cyangain	58	image.color.p7.custom.cyanlum	58
image.color.p7.custom.cyanx	58	image.color.p7.custom.cyany	58
image color n7 custom gainsavailable	58	image color n7 custom greengain	59

image.connector.l2displayporta.colorprimariesavailable

94

image.connector.l2displayporta.colorprimaries

optics.zoom.maxposition

optics.zoom.position

144

144

144

144

optics.zoom.location

optics.zoom.minposition

optics.zoom.safetocalibrate	144	optics.zoom.safetooperate	145
optics.zoom.state	145	optics.zoom.target	145
peripheral.frame.horizontal.calibrationstate	145	peripheral.frame.horizontal.state	146
peripheral.frame.rotation.calibrationstate	146	peripheral.frame.rotation.state	146
peripheral.frame.vertical.calibrationstate	147	peripheral.frame.vertical.state	147
remotecontrol.address	147	remotecontrol.broadcastaddress	147
remotecontrol.sensors.front.enable	148	remotecontrol.sensors.front.name	148
remotecontrol.sensors.rear.enable	148	remotecontrol.sensors.rear.name	148
remotecontrol.sensors.side.enable	149	remotecontrol.sensors.side.name	149
screen.hdrboost	149	screen.luminance	149
statistics.laserruntime.value	150	statistics.laserstrikes.value	150
statistics.projectorruntime.value	150	statistics.systemtime.value	150
statistics.uptime.value	150	system.articlenumber	151
system.colorwheel	151	system.colorwheelname	151
system.eco.available	151	system.eco.enable	151
system.error.timeout.duration	151	system.error.timeout.enable	152
system.familyname	152	system.firmwareversion	152
system.initialstate	152	system.license.applicable	153
system.license.available	153	system.license.option.flexbrightness.enabled	153
system. license. option. flexbrightness. maximum lightout put	153	system. license. option. flex brightness. maximum light output attempts left	153
system. license. option. flex brightness. maximum light outputs	154	system.license.options	154
system.license.valid	154	system.modelname	154
system.on.timeout.duration	154	system.on.timeout.enable	155
system.ready.timeout.duration	155	system.ready.timeout.enable	155
system.serialnumber	155	system.standby.available	155
system.standby.enable	156	system.standby.timeout.duration	156
system.standby.timeout.enable	156	system.state	156
ui.access.enduser	157	ui.backlight.state	157
ui.backlight.timeout	157	ui.hasstealthmode	157
ui.keyboardshortcut	157	ui.layer.adjustment.enable	158
ui.layer.adjustment.icon	158	ui.layer.adjustment.layerposition	159
ui.layer.adjustment.rangedvalue	159	ui.layer.adjustment.showleftrightbuttons	159
ui.layer.adjustment.showupdownbuttons	160	ui.layer.advancedblend.drawing	160
ui.layer.advancedblend.enable	160	ui.layer.advancedblend.palette	160
ui.layer.basicblacklevel.color	160	ui.layer.basicblacklevel.enable	161
ui.layer.basicblacklevel.selection	161	ui.layer.basicblacklevel.selectioncolor	161
ui.layer.basicblend.color	161	ui.layer.basicblend.enable	161
ui.layer.basicblend.selection	162	ui.layer.basicblend.selectioncolor	162
ui.layer.fourcorner.cornercolor	162	ui.layer.fourcorner.enable	162
ui.layer.fourcorner.linecolor	162	ui.layer.fourcorner.lines	163
ui.layer.fourcorner.selection	163	ui.layer.fourcorner.selectioncolor	163
ui.layer.grid.color	163	ui.layer.grid.enable	164
ui.layer.grid.lines	164	ui.layer.grid.mark	164
ui.layer.grid.points	164	ui.layer.grid.showlines	165

Pulse API	Reference G	uide	Version 1.7
ui.layer.grid.showpoints	165	ui.menu	165
ui.menuposition	165	ui.minimize	166
ui.notificationfiltercodes	166	ui.notificationfilterseverity	166
ui.osd	166	ui.poweroffhint	166
ui.sourcesignal	167	ui.sourcesignalposition	167
ui.stealthmode	167		

Method index

dmx.listchannels	169	dmx.listmodes	169
environment.getalarminfo	169	environment.getcontrolblocks	170
illumination.clo.engage	171	illumination.laser.getserialnumber	171
image.color.p7.custom.copypresettocustom	171	image.color.p7.custom.resetpreset	172
image.color.p7.custom.resettonative	172	image.color.rgbmode.nextrgbmode	172
image.connector.l1displayport.edid.list	172	image.connector.l1hdbaset1.edid.list	173
image.connector.l1hdbaset2.edid.list	173	image.connector.l1hdmi.edid.list	173
image.connector.l2displayporta.edid.list	174	image.connector.l2displayportb.edid.list	174
image.connector.l2displayportc.edid.list	174	image.connector.l2displayportd.edid.list	175
image.connector.list	175	image.display.listdisplaymodes	175
image.processing.blacklevel.basicblacklevel.getblacklevelarea	175	image.processing.blacklevel.basicblacklevel.getwarpedblacklevelarea	177
image.processing.blacklevel.file.delete	178	image.processing.blacklevel.file.list	178
image.processing.blend.basicblend.getblendarea	178	image.processing.blend.basicblend.getwarpedblendarea	181
image.processing.blend.file.delete	183	image.processing.blend.file.list	183
image.processing.warp.file.delete	183	image.processing.warp.file.list	183
image.processing.warp.fourcorners.getscaledcorners	184	image.processing.warp.warpscaledpoints	184
image.processing.warpgrid.getgrid	185	image.processing.warpgrid.getgridsize	185
image.processing.warpgrid.getscaledgrid	185	image.resolution.list	186
image.source.l1displayport.listconnectors	186	image.source.l1hdbaset1.listconnectors	187
image.source.l1hdbaset2.listconnectors	187	image.source.l1hdmi.listconnectors	187
image.source.l1quadsdi.listconnectors	188	image.source.l1sdia.listconnectors	188
image.source.l1sdib.listconnectors	189	image.source.l1sdic.listconnectors	189
image.source.l1sdid.listconnectors	190	image.source.l2displayporta.listconnectors	190
image.source.l2displayportb.listconnectors	190	image.source.l2displayportc.listconnectors	191
image.source.l2displayportd.listconnectors	191	image.source.l2dualdpab.listconnectors	192
image.source.l2dualdpac.listconnectors	192	image.source.l2dualdpbd.listconnectors	193
image.source.l2dualdpcd.listconnectors	193	image.source.l2dualheaddpac.listconnectors	193
image.source.l2dualheaddpbd.listconnectors	194	image.source.l2dualheaddualdpabcd.listconnectors	194
image.source.l2quadcolumndp.listconnectors	195	image.source.l2quaddp.listconnectors	195
image.source.list	196	image.stereo.listdarktime	196
image.testpattern.file.delete	196	image.testpattern.file.list	196
image.testpattern.list	196	image.testpattern.setproperties	197
image.window.list	197	keydispatcher.sendclickevent	198
keydispatcher.sendpressevent	199	keydispatcher.sendreleaseevent	201
led.activity	202	led.list	202
lightmeasurement.getlightoutput	202	network.list	203
notification.dismiss	203	notification.list	203
notification.listsuppressed	204	notification.log	204
notification.suppress	205	notification.unsuppress	206
notification.unsuppressall	206	optics.focus.addlocation	206
optics.focus.calibrate	206	optics.focus.runforward	206
optics.focus.runforwardtime	207	optics.focus.runreverse	207

Pulse API	Reference G	Guide	Version 1.7
optics.focus.runreversetime	207	optics.focus.setlocation	207
optics.focus.stepforward	208	optics.focus.stepreverse	208
optics.focus.stop	208	optics.getvalidlensids	208
optics.lensshift.horizontal.addlocation	209	optics.lensshift.horizontal.calibrate	209
optics.lensshift.horizontal.runforward	209	optics.lensshift.horizontal.runforwardtime	209
optics.lensshift.horizontal.runreverse	209	optics.lensshift.horizontal.runreversetime	210
optics.lensshift.horizontal.setlocation	210	optics.lensshift.horizontal.stepforward	210
optics.lensshift.horizontal.stepreverse	210	optics.lensshift.horizontal.stop	211
optics.lensshift.vertical.addlocation	211	optics.lensshift.vertical.calibrate	211
optics.lensshift.vertical.runforward	211	optics.lensshift.vertical.runforwardtime	211
optics.lensshift.vertical.runreverse	212	optics.lensshift.vertical.runreversetime	212
optics.lensshift.vertical.setlocation	212	optics.lensshift.vertical.stepforward	212
optics.lensshift.vertical.stepreverse	212	optics.lensshift.vertical.stop	213
optics.setlensid	213	optics.shifttocenter	213
optics.shutter.getobjectpath	213	optics.shutter.toggle	214
optics.zoom.addlocation	214	optics.zoom.calibrate	214
optics.zoom.runforward	214	optics.zoom.runforwardtime	214
optics.zoom.runreverse	215	optics.zoom.runreversetime	215
optics.zoom.setlocation	215	optics.zoom.stepforward	215
optics.zoom.stepreverse	215	optics.zoom.stop	216
peripheral.frame.horizontal.calibrate	216	peripheral.frame.horizontal.runforward	216
peripheral.frame.horizontal.runreverse	216	peripheral.frame.horizontal.stepforward	216
peripheral.frame.horizontal.stepreverse	217	peripheral.frame.horizontal.stop	217
peripheral.frame.rotation.calibrate	217	peripheral.frame.rotation.runforward	217
peripheral.frame.rotation.runreverse	217	peripheral.frame.rotation.stepforward	218
peripheral.frame.rotation.stepreverse	218	peripheral.frame.rotation.stop	218
peripheral.frame.vertical.calibrate	218	peripheral.frame.vertical.runforward	218
peripheral.frame.vertical.runreverse	219	peripheral.frame.vertical.stepforward	219
peripheral.frame.vertical.stepreverse	219	peripheral.frame.vertical.stop	219
remotecontrol.listsensors	219	statistics.laserruntime.getname	220
statistics.laserruntime.getunit	220	statistics.laserstrikes.getname	220
statistics.laserstrikes.getunit	220	statistics.listcounters	221
statistics.projectorruntime.getname	221	statistics.projectorruntime.getunit	222
statistics.systemtime.getname	222	statistics.systemtime.getunit	222
statistics.uptime.getname	223	statistics.uptime.getunit	223
system.activity	223	system.boards.getboardinfo	223
system.boards.getboardlist	224	system.boards.getdeviceinfo	224
system.boards.getmissingboardlist	224	system.boards.getmoduleinfo	225
system.getidentification	225	system.getidentifications	225
system.getsystemdate	226	system.gotoeco	226
system.gotoready	226	system.license.option.flexbrightness.getmaximumlightoutputce	ode 226
system. license. option. flex brightness. set maximum light output	227	system.license.option.flexbrightness.setmaximumlightoutputce	ode 227
system.listresetdomains	227	system.poweroff	228
system.poweron	228	system.reset	228
Dulas ADI	D-f	N.: J.	050 -{ 050

Pulse API	Reference	Guide	Version 1.7
system.resetall	229	ui.settings.get	230
ui.settings.getfonticons	230	ui.settings.geticons	231
ui.settings.keys	231	ui.settings.list	231
ui.settings.remove	232	ui.settings.set	232
ui.togglestealthmode	232		

Signal index

modelupdated	233	image.connector.l1displayport.edid.listchanged	233
image.connector.l1hdbaset1.edid.listchanged	233	image.connector.l1hdbaset2.edid.listchanged	233
image.connector.l1hdmi.edid.listchanged	233	image.connector.l2displayporta.edid.listchanged	234
image. connector. I2 display port b. edid. list changed	234	image.connector.l2displayportc.edid.listchanged	234
image. connector. I2 display port d. edid. list changed	234	image.processing.blacklevel.file.listchanged	234
image.processing.blend.file.listchanged	234	image.processing.warp.file.listchanged	234
image.processing.warpgrid.changed	235	image.processing.warpgrid.gridchanged	235
image.testpattern.added	235	image.testpattern.changed	235
image.testpattern.file.listchanged	236	image.testpattern.removed	236
network.added	236	network.removed	236
notification.dismissed	237	notification.emitted	237
system.identificationchanged	238	system.license.licensechanged	238
system.performed	238	ui.settings.added	239
ui.settings.changed	239	ui.settings.removed	239

File index

image.connector.edid.transfer	240	image.processing.blacklevel.file.transfer	240
image.processing.blend.file.transfer	241	image.processing.warp.file.transfer	241
image.testpattern.file.transfer	242	notification.logger.transfer	242