

RS232 and Network Command Catalog

For JSON RPC/Pulse Based Projectors

For I600

End User

Reference guide

2.5

2024-06-15

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	8
Signals	11
Notifications	13
Introspection API	14
Object changed signal	17
File endpoints	18
Programmers guide	20
Best practice	20
Basic operation	20
Sources	23
Illumination	29
Picture settings	33
Warping with grid files	36
Blending with images	37
Black level adjustment with images	39
Environment information	40
ECO mode	43
Important note about the API documentation	43
Properties	44
Methods	225
Signals	321
Files	330
Property index	335
Method index	344
Signal index	347
File index	348

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.

Discovering a projector on the local network

The projector can be discovered by using the simple search and discovery protocol (SSDP). To discover only the projectors on the network, use the following search:

```
"urn:schemas-upnp-org:device:projector:1"
```

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

Request

```
{
  "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 `tempctrl.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 Pyeongchang {"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
<pre>{ "jsonrpc": "2.0", "method": "authenticate", "params": { "code": 98765 }, "id": 1 }</pre>	<pre>{ "jsonrpc": "2.0", "result": true, "id": 1 }</pre>

Service API

Property	Type	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
<pre>{ "jsonrpc": "2.0", "method": "ledctrl.blink", "params": { "led": "systemstatus", "color": "red", "period": 42 }, "id": 3 }</pre>	<pre>{ "jsonrpc": "2.0", "result": 0, "id": 3 }</pre>

Properties

API for setting and getting property values

Set value of a property

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "property.set", "params": { "property": "objectname.propertyname", "value": 100 }, "id": 3 }</pre>	<pre>{ "jsonrpc": "2.0", "result": true, "id": 3 }</pre>

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
<pre>{ "jsonrpc": "2.0", "method": "property.get", "params": { "property": "objectname.propertyname" }, "id": 4 }</pre>	<pre>{ "jsonrpc": "2.0", "result": 100, "id": 4 }</pre>

Read values of multiple properties

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "property.get", "params": { "property": ["image.brightness", "image.contrast"] }, "id": 5 }</pre>	<pre>{ "jsonrpc": "2.0", "result": { "image.brightness": 0, "image.contrast": 1 }, "id": 5 }</pre>

Observe changes on one property

For change notifications, see [Notifications](#)

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "property.subscribe", "params": { "property": "image.brightness" }, "id": 6 }</pre>	<pre>{ "jsonrpc": "2.0", "result": true, "id": 6 }</pre>

Observe changes on multiple properties

For change notifications, see [Notifications](#)

Request

Response

<pre>{ "jsonrpc": "2.0", "method": "property.subscribe", "params": { "property": ["image.brightness", "image.contrast"] }, "id": 7 }</pre>	<pre>{ "jsonrpc": "2.0", "result": true, "id": 7 }</pre>
--	--

Stop observing one property

For change notifications, see [Notifications](#)

Request

Response

<pre>{ "jsonrpc": "2.0", "method": "property.unsubscribe", "params": { "property": "image.brightness" }, "id": 8 }</pre>	<pre>{ "jsonrpc": "2.0", "result": true, "id": 8 }</pre>
--	--

Stop observing multiple properties

For change notifications, see [Notifications](#)

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "property.unsubscribe", "params": { "property": ["image.brightness", "image.contrast"] }, "id": 9 }</pre>	<pre>{ "jsonrpc": "2.0", "result": true, "id": 9 }</pre>

Signals

Subscribe to a signal

For change notifications, see [Notifications](#)

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "signal.subscribe", "params": { "signal": "modelupdated" }, "id": 10 }</pre>	<pre>{ "jsonrpc": "2.0", "result": true, "id": 10 }</pre>

Subscribe to multiple signals

For change notifications, see [Notifications](#)

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "signal.subscribe", "params": { "signal": ["modelupdated", "image.processing.warp.gridchanged"] }, "id": 11 }</pre>	<pre>{ "jsonrpc": "2.0", "result": true, "id": 11 }</pre>

Unsubscribe from a signal

For change notifications, see [Notifications](#)

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "signal.unsubscribe", "params": { "signal": "modelupdated" }, "id": 12 }</pre>	<pre>{ "jsonrpc": "2.0", "result": true, "id": 12 }</pre>

Unsubscribe from multiple signals

For change notifications, see [Notifications](#)

Request

Response

```
{
  "jsonrpc": "2.0",
  "method": "signal.unsubscribe",
  "params": {
    "signal": [
      "modelupdated",
      "image.processing.warp.gridchanged"
    ]
  },
  "id": 13
}
```

```
{
  "jsonrpc": "2.0",
  "result": true,
  "id": 13
}
```

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

```
{
  "jsonrpc": "2.0",
  "method": "property.changed",
  "params": {
    "property": [
      {
        "objectname.propertyname": 100
      },
      {
        "otherobject.otherproperty": "bar"
      }
    ]
  }
}
```

Signals

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

Notification

```
{
  "jsonrpc": "2.0",
  "method": "signal.callback",
  "params": {
    "signal": [
      {
        "objectname.signalname": {
          "arg1": 100,
          "arg2": "cat"
        }
      },
      {
        "otherobject.othersignal": {
          "foo": "bar"
        }
      }
    ]
  }
}
```

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)

Request

Response

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

- or -

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

Introspection API (non recursive)

Request

Response

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

- or -

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

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

Property	Type	Required	Comments
	<code>{"recursive": bool}</code>	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
<pre>{ "jsonrpc": "2.0", "method": "signal.subscribe", "params": { "signal": "modelupdated" }, "id": 2 }</pre>	<pre>{ "jsonrpc": "2.0", "result": true, "id": 2 }</pre>

Callback method on client side

Notification

```
{
  "jsonrpc": "2.0",
  "method": "signal.callback",
  "params": {
    "signal": [
      {
        "introspect.objectchanged": {
          "object": "motors.motor1",
          "newobject": true
        }
      }
    ]
  }
}
```

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 `curl` 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.

Best practice

This chapter describes some basic guidelines and best practice when communicating with the projector.

Availability of information

Properties, methods and signals are not necessarily available at all times. This is because the projector has different states. One of these states is **standby**. In this state, not all the projector services are running. This means that properties and methods that are available when the projector is completely on, are not necessarily available when in standby. For example, the shutter, which is available when in **ready** or **on**, is not available in **standby**.

Tip! Do not poll properties or call methods that are not available.

Polling properties, typically by calling `property.get` at regular intervals, with a property that's currently unavailable, just creates overhead in the system. Doing this will log an internal error and also return an error to the caller.

Setting property values

Sometimes, setting a property value can actually trigger big operations in the projector, causing the projector to be busy for some time. For example, setting a color coordinate of the **RealColor P7** properties may take a while for the projector to process.

Tip! Always wait for the confirmation before setting the property value again.

When calling `property.set` always specify an `id`. Doing so means that the projector will respond with the result of the call. Not doing so, means that you will not be notified when the projector is finished with processing the request. When the projector response is received, it is safe to again set the property value. Not waiting for the result, and quickly calling multiple `property.set` may lead to unexpected results.

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
<pre>{ "jsonrpc": "2.0", "method": "property.get", "params": { "property": "system.state" }, "id": 1 }</pre>	<pre>{ "jsonrpc": "2.0", "result": "on", "id": 1 }</pre>

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.

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "property.subscribe", "params": { "property": "system.state" }, "id": 2 }</pre>	<pre>{ "jsonrpc": "2.0", "result": true, "id": 2 }</pre>

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

Notification

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

Power on

To power on the projector, issue the following request.

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "system.poweron", "params": { "property": "system.state" }, "id": 3 }</pre>	<pre>{ "jsonrpc": "2.0", "result": null, "id": 3 }</pre>

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	Response
<pre>{ "jsonrpc": "2.0", "method": "system.poweroff", "params": { "property": "system.state" }, "id": 4 }</pre>	<pre>{ "jsonrpc": "2.0", "result": null, "id": 4 }</pre>

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
<pre>{ "jsonrpc": "2.0", "method": "property.get", "params": { "property": "image.window.main.source" }, "id": 0 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 0, "result": "DisplayPort 1" }</pre>

List available sources

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

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "image.source.list", "id": 1 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 1, "result": ["DVI 1", "DVI 2", "DisplayPort 1", "DisplayPort 2", "Dual DVI", "Dual DisplayPort", "Dual Head DVI", "Dual Head DisplayPort", "HDBaseT", "HDMI", "SDI"] }</pre>

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
<pre>{ "jsonrpc": "2.0", "method": "property.set", "params": { "property": "image.window.main.source", "value": "DisplayPort 1" }, "id": 2 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 2, "result": true }</pre>

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

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 **DisplayPort 1**, remove all non word characters and convert to all lower case characters. This can be done quite easy by using regular expressions. Example in **JavaScript** 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

```
{
  "jsonrpc": "2.0",
  "method": "image.source.displayport1.listconnectors",
  "id": 4
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 4,
  "result": [
    {
      "gridposition": {
        "row": 0,
        "column": 0,
        "plane": 0
      },
      "name": "DisplayPort 1"
    }
  ]
}
```

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 `DisplayPort 1`, this translated to a connector name of `displayport1`. We can then get signal information by using the following request.

Request

```
{
  "jsonrpc": "2.0",
  "method": "property.get",
  "params": {
    "property":
      "image.connector.displayport1.detectedsignal"
  },
  "id": 5
}
```

Response

```
{
  "jsonrpc": "2.0",
  "id": 5,
  "result": {
    "active": true,
    "name": "2560x1600 @ 50.10Hz",
    "vertical_total": 1638,
    "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 for changes on a number of properties.

Subscribe to the source property of the window

Request

Response

<pre>{ "jsonrpc": "2.0", "method": "property.subscribe", "params": { "property": "image.window.main.source" }, "id": 6 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 6, "result": true }</pre>
--	--

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** server will send notifications as shown below.

Notification

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

Notification

```
{
  "jsonrpc": "2.0",
  "method": "property.changed",
  "params": {
    "property": [
      {
        "image.window.main.source": "DisplayPort 2"
      }
    ]
  }
}
```

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 source must 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
 - Call `image.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.

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "property.get", "params": { "property": "illumination.state" }, "id": 0 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 0, "result": "Off" }</pre>

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
<pre>{ "jsonrpc": "2.0", "method": "property.subscribe", "params": { "property": "illumination.state" }, "id": 1 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 1, "result": true }</pre>

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
<pre>{ "jsonrpc": "2.0", "method": "property.changed", "params": { "property": [{ "illumination.state": "On" }] } }</pre>

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
<pre>{ "jsonrpc": "2.0", "method": "introspect", "params": { "property": "illumination.sources", "recursive": false }, "id": 2 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 2, "result": { "objects": [{ "name": "illuminations.sources.laser" }] } }</pre>

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
<pre>{ "jsonrpc": "2.0", "method": "property.get", "params": { "property": "illumination.sources.laser.power" }, "id": 3 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 3, "result": 30 }</pre>

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

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "property.subscribe", "params": { "property": ["illumination.sources.laser.power"] }, "id": 4 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 4, "result": true }</pre>

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

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "property.set", "params": { "property": "illumination.sources.laser.power", "value": 40 }, "id": 5 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 5, "result": true }</pre>

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

Notification

```
{
  "jsonrpc": "2.0",
  "method": "property.changed",
  "params": {
    "property": [
      {
        "illumination.sources.laser.power": 40
      }
    ]
  }
}
```

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

Request

Response

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

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.

Request

Response

<pre>{ "jsonrpc": "2.0", "method": "introspect", "params": { "object": "image", "recursive": false }, "id": 6 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 6, "result": { "name": "image", "properties": [{ "name": "brightness", "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" }] } }</pre>
---	---

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
<pre>{ "jsonrpc": "2.0", "method": "property.get", "params": { "property": "image.brightness" }, "id": 7 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 7, "result": 0 }</pre>

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

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "property.subscribe", "params": { "property": ["image.brightness"] }, "id": 8 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 8, "result": true }</pre>

The set the brightness value, use the following request.

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "property.set", "params": { "property": "image.brightness", "value": 0.15 }, "id": 9 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 9, "result": true }</pre>

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

Notification

```
{
  "jsonrpc": "2.0",
  "method": "property.changed",
  "params": {
    "property": [
      {
        "image.brightness": 0.15
      }
    ]
  }
}
```

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
<pre>{ "jsonrpc": "2.0", "method": "property.set", "params": { "property": "image.processing.warp.enable", "value": true }, "id": 10 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 10, "result": true }</pre>

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
<pre>{ "jsonrpc": "2.0", "method": "property.set", "params": { "property": "image.processing.warp.file.selected", "value": "warp.xml" }, "id": 11 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 11, "result": true }</pre>

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

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "property.set", "params": { "property": "image.processing.warp.file.enable", "value": true }, "id": 12 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 12, "result": true }</pre>

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
WQXGA+ (2716 x 1600)	1358 x 800
4K	1280 x 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
<pre>{ "jsonrpc": "2.0", "method": "property.set", "params": { "property": "image.processing.blend.file.selected", "value": "mask.png" }, "id": 13 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 13, "result": true }</pre>

To enable the blend mask use the following request.

Request	Response
<pre>{ "jsonrpc": "2.0", "method": "property.set", "params": { "property": "image.processing.blend.file.enable", "value": true }, "id": 14 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 14, "result": true }</pre>

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
WQXGA+ (2716 x 1600)	1358 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

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

Response

```
{
  "jsonrpc": "2.0",
  "id": 15,
  "result": true
}
```

To enable the black level mask use the following request.

Request

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

Response

```
{
  "jsonrpc": "2.0",
  "id": 16,
  "result": true
}
```

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

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

Request

Response

<pre>{ "jsonrpc": "2.0", "method": "environment.getcontrolblocks", "params": { "type": "Sensor", "valuetype": "Temperature" }, "id": 18 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 18, "result": { "environment.laser.board0.bank0.temperature": 35.3, "environment.laser.board0.bank1.temperature": 34.8, "environment.laser.board0.bank2.temperature": 35.3, "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, "environment.temperature.outlet": 29.4, "environment.temperature.scalerfpga": 52.8 } }</pre>
---	---

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.

Fan speeds

To get fan speeds, use the following request.

Request

Response

<pre>{ "jsonrpc": "2.0", "method": "environment.getcontrolblocks", "params": { "type": "Sensor", "valuetype": "Speed" }, "id": 19 }</pre>	<pre>{ "jsonrpc": "2.0", "id": 19, "result": { "environment.fan.ar1.tacho": 1800, "environment.fan.ar2.tacho": 1850, "environment.fan.ar3.tacho": 1750, "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 } }</pre>
---	---

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:

```
:POWER1\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

backup.auto.error

The error message in case of a backup error state.

MODELS

 All

Access: R

Name	Type
error	string

backup.auto.ipaddress

Initial IP address for backup actions.

MODELS

 All

Access: R

Name	Type
ipaddress	string

backup.auto.outofsyncdate

Date on which SyncState became Asynchronous

MODELS

 All

Access: R

Name	Type
outofsyncdate	string

backup.auto.state

The current state of the automatic backup

MODELS

 All

Access: R

Name	Type
state	enum
Values	
"Inactive"	
"Active"	
"RestoringFull"	
"RestoringHistory"	

"Discarding"

"Postponing"

"Error"

backup.auto.syncstate

Whether the backup is in sync

MODELS All

Access: R

Name	Type
syncstate	enum
Values	
"Synchronous"	
"Asynchronous"	

diagnostics.selftest.progress

Progress for the selftest.

MODELS All

Access: R

Name	Type
progress	{ }
NameType	
base	int
hints	{ }
NameType	

diagnostics.selftest.result

The result and overview of all executed selftests

MODELS All

Access: R

Name	Type
result	{ }
NameType	
status	enum
Values	
"Idle"	
"Busy"	
"Success"	
"Failure"	
"Aborted"	

```
executiontime      string
executionduration  { }
```

Name	Type
base	int
hints	{ }

```
testResults        [ { } ]
                    { }
```

Name	Type
name	string
status	enum

Values
"Idle"
"Busy"
"Success"
"Failure"
"Aborted"

```
executiontime      string
executionduration  { }
```

Name	Type
base	int
hints	{ }

```
info               [ { } ]
                    { }
```

Name	Type
timestamp	string
severity	enum

Values
"Info"
"Warn"
"Error"

```
description  string
```

diagnostics.selftest.status

The current status of the executed test-list

MODELS

 All

Access: R

Name	Type
status	enum
Values	

"Idle"

"Busy"

"Success"

"Failure"

"Aborted"

dmx.artnet

Artnet enabled or not

MODELSAll

Access: RW

Name	Type
artnet	bool

dmx.artnetnet

Artnet net selection

MODELSAll

Access: RW

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

dmx.artnetuniverse

Artnet universe selection

MODELSAll

Access: RW

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

dmx.mode

Current mode

MODELS

 All

Access: RW

Name	Type
mode	string

dmx.monitor.channel01.function

Description for the dmx channel

MODELS

 All

Access: R

Name	Type
function	string

dmx.monitor.channel01.offset

Offset of the channel.

MODELS

 All

Access: R

Name	Type						
offset	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

dmx.monitor.channel01.value

Current dmx value for the channel

MODELS

 All

Access: R

Name	Type						
value	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

dmx.monitor.channel02.function

Description for the dmx channel

MODELS

 All

Access: R

Name	Type
function	string

dmx.monitor.channel02.offset

Offset of the channel.

MODELS All

Access: R

Name	Type						
offset	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						

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

dmx.monitor.channel02.value

Current dmx value for the channel

MODELS All

Access: R

Name	Type						
value	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						

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

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.

MODELS All

Access: R

Name	Type
active	bool

dmx.shutdown

Shutdown enabled or not

MODELS All

Access: RW

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

shutdown

bool

dmx.shutdowntimeout

Time out for shutdown in minutes.

MODELS

All

Access: RW

Name	Type
shutdowntimeout	int
Constraints	
Minimum	1
Maximum	60
Step size	1
Precision	1

dmx.startchannel

The dmx start channel [1..512].

MODELS

All

Access: RW

Name	Type
startchannel	int
Constraints	
Minimum	1
Maximum	512
Step size	1
Precision	1

environment.alarmstate

[DEPRECATED] Replaced by 'system.health'

MODELS

All

Access: R

Name	Type
alarmstate	enum
Values	
"Error"	
"Alert"	
"Warning"	
"Ok"	

firmware.availablepackages

The firmware packages available for upgrade

MODELS

All

Access: R

Name	Type								
availablepackages	[{ }]								
	{ }								
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>name</td><td>string</td></tr><tr><td>version</td><td>string</td></tr><tr><td>source</td><td>enum</td></tr></table>	Name	Type	name	string	version	string	source	enum
Name	Type								
name	string								
version	string								
source	enum								
	<table><tr><th>Values</th></tr><tr><td>"USB"</td></tr><tr><td>"SD"</td></tr></table>	Values	"USB"	"SD"					
Values									
"USB"									
"SD"									

firmware.component.backplane-boot-redstar.actualversion

The version of the currently installed firmware.

MODELS

All

Access: R

Name	Type
actualversion	string

firmware.component.backplane-boot-redstar.displayname

The user-friendly name of the firmware component.

MODELS

All

Access: R

Name	Type
displayname	string

firmware.component.backplane-boot-redstar.error

The error message in case of an upgrade error.

MODELS

All

Access: R

Name	Type
error	string

firmware.component.backplane-boot-redstar.mismatch

Indicate whether there is a version mismatch

MODELS

All

Access: R

Name	Type
mismatch	bool

firmware.component.backplane-boot-redstar.progress

The progress of the current firmware upgrade

MODELS All

Access: R

Name	Type
progress	{ }
Name	Type
base	int
hints	{ }
Name	Type

firmware.component.backplane-boot-redstar.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS All

Access: R

Name	Type
requiredversion	string

firmware.component.backplane-boot-redstar.status

The status of the current firmware upgrade

MODELS All

Access: R

Name	Type
status	enum
Values	
"Inactive"	
"Active"	
"Error"	
"Finished"	

firmware.component.backplane-redstar-fpga.actualversion

The version of the currently installed firmware.

MODELS All

Access: R

Name	Type
actualversion	string

firmware.component.backplane-redstar-fpga.displayname

The user-friendly name of the firmware component.

MODELS All

Access: R

Name	Type
displayname	string

firmware.component.backplane-redstar-fpga.error

The error message in case of an upgrade error.

MODELS All

Access: R

Name	Type
error	string

firmware.component.backplane-redstar-fpga.mismatch

Indicate whether there is a version mismatch

MODELS All

Access: R

Name	Type
mismatch	bool

firmware.component.backplane-redstar-fpga.progress

The progress of the current firmware upgrade

MODELS All

Access: R

Name	Type						
progress	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

firmware.component.backplane-redstar-fpga.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS All

Access: R

Name	Type
requiredversion	string

firmware.component.backplane-redstar-fpga.status

The status of the current firmware upgrade

MODELS All

Access: R

Name	Type
status	enum
Values	
"Inactive"	
"Active"	
"Error"	
"Finished"	

firmware.component.backplane-run-redstar.actualversion

The version of the currently installed firmware.

MODELS All

Access: R

Name	Type
actualversion	string

firmware.component.backplane-run-redstar.displayname

The user-friendly name of the firmware component.

MODELS All

Access: R

Name	Type
displayname	string

firmware.component.backplane-run-redstar.error

The error message in case of an upgrade error.

MODELS All

Access: R

Name	Type
error	string

firmware.component.backplane-run-redstar.mismatch

Indicate whether there is a version mismatch

MODELS

 All

Access: R

Name	Type
mismatch	bool

firmware.component.backplane-run-redstar.progress

The progress of the current firmware upgrade

MODELS

 All

Access: R

Name	Type						
progress	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

firmware.component.backplane-run-redstar.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS

 All

Access: R

Name	Type
requiredversion	string

firmware.component.backplane-run-redstar.status

The status of the current firmware upgrade

MODELS

 All

Access: R

Name	Type					
status	enum					
	<table><tr><th>Values</th></tr><tr><td>"Inactive"</td></tr><tr><td>"Active"</td></tr><tr><td>"Error"</td></tr><tr><td>"Finished"</td></tr></table>	Values	"Inactive"	"Active"	"Error"	"Finished"
Values						
"Inactive"						
"Active"						
"Error"						
"Finished"						

firmware.component.components

List of components

MODELS

 All

Access: R

Name	Type
components	[string]

firmware.component.formatter-1 dlp-fpga.actualversion

The version of the currently installed firmware.

MODELS

 All

Access: R

Name	Type
actualversion	string

firmware.component.formatter-1 dlp-fpga.displayname

The user-friendly name of the firmware component.

MODELS

 All

Access: R

Name	Type
displayname	string

firmware.component.formatter-1 dlp-fpga.error

The error message in case of an upgrade error.

MODELS

 All

Access: R

Name	Type
error	string

firmware.component.formatter-1 dlp-fpga.mismatch

Indicate whether there is a version mismatch

MODELS

 All

Access: R

Name	Type
mismatch	bool

firmware.component.formatter-1 dlp-fpga.progress

The progress of the current firmware upgrade

MODELS

 All

Access: R

Name	Type						
progress	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						

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

firmware.component.formatter-1 dlp-fpga.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS All

Access: R

Name	Type
requiredversion	string

firmware.component.formatter-1 dlp-fpga.status

The status of the current firmware upgrade

MODELS All

Access: R

Name	Type					
status	enum					
	<table><tr><th>Values</th></tr><tr><td>"Inactive"</td></tr><tr><td>"Active"</td></tr><tr><td>"Error"</td></tr><tr><td>"Finished"</td></tr></table>	Values	"Inactive"	"Active"	"Error"	"Finished"
Values						
"Inactive"						
"Active"						
"Error"						
"Finished"						

firmware.component.formatter-ddp442x-sequences.actualversion

The version of the currently installed firmware.

MODELS All

Access: R

Name	Type
actualversion	string

firmware.component.formatter-ddp442x-sequences.displayname

The user-friendly name of the firmware component.

MODELS All

Access: R

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

displayname

string

firmware.component.formatter-ddp442x-sequences.error

The error message in case of an upgrade error.

MODELS

All

Access: R

Name	Type
error	string

firmware.component.formatter-ddp442x-sequences.mismatch

Indicate whether there is a version mismatch

MODELS

All

Access: R

Name	Type
mismatch	bool

firmware.component.formatter-ddp442x-sequences.progress

The progress of the current firmware upgrade

MODELS

All

Access: R

Name	Type						
progress	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

firmware.component.formatter-ddp442x-sequences.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS

All

Access: R

Name	Type
requiredversion	string

firmware.component.formatter-ddp442x-sequences.status

The status of the current firmware upgrade

MODELS

All

Access: R

Name	Type
status	enum
Values	
"Inactive"	
"Active"	
"Error"	
"Finished"	

firmware.component.formatter-rallar-boot.actualversion

The version of the currently installed firmware.

MODELS All

Access: R

Name	Type
actualversion	string

firmware.component.formatter-rallar-boot.displayname

The user-friendly name of the firmware component.

MODELS All

Access: R

Name	Type
displayname	string

firmware.component.formatter-rallar-boot.error

The error message in case of an upgrade error.

MODELS All

Access: R

Name	Type
error	string

firmware.component.formatter-rallar-boot.mismatch

Indicate whether there is a version mismatch

MODELS All

Access: R

Name	Type
mismatch	bool

firmware.component.formatter-rallar-boot.progress

The progress of the current firmware upgrade

MODELS

All

Access: R

Name	Type						
progress	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

firmware.component.formatter-rallar-boot.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS

All

Access: R

Name	Type
requiredversion	string

firmware.component.formatter-rallar-boot.status

The status of the current firmware upgrade

MODELS

All

Access: R

Name	Type
status	enum
	Values "Inactive" "Active" "Error" "Finished"

firmware.component.formatter-rallar-run.actualversion

The version of the currently installed firmware.

MODELS

All

Access: R

Name	Type
actualversion	string

firmware.component.formatter-rallar-run.displayname

The user-friendly name of the firmware component.

MODELS

All

Access: R

Name	Type
displayname	string

firmware.component.formatter-rallar-run.error

The error message in case of an upgrade error.

MODELS

All

Access: R

Name	Type
error	string

firmware.component.formatter-rallar-run.mismatch

Indicate whether there is a version mismatch

MODELS

All

Access: R

Name	Type
mismatch	bool

firmware.component.formatter-rallar-run.progress

The progress of the current firmware upgrade

MODELS

All

Access: R

Name	Type
progress	{ }
Name	Type
base	int
hints	{ }
Name	Type

firmware.component.formatter-rallar-run.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS

All

Access: R

Name	Type
requiredversion	string

firmware.component.formatter-rallar-run.status

The status of the current firmware upgrade

MODELS All

Access: R

Name	Type
status	enum
Values	
"Inactive"	
"Active"	
"Error"	
"Finished"	

firmware.component.green-controller-boot-v1.actualversion

The version of the currently installed firmware.

MODELS All

Access: R

Name	Type
actualversion	string

firmware.component.green-controller-boot-v1.displayname

The user-friendly name of the firmware component.

MODELS All

Access: R

Name	Type
displayname	string

firmware.component.green-controller-boot-v1.error

The error message in case of an upgrade error.

MODELS All

Access: R

Name	Type
error	string

firmware.component.green-controller-boot-v1.mismatch

Indicate whether there is a version mismatch

MODELS All

Access: R

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

mismatch bool

firmware.component.green-controller-boot-v1.progress

The progress of the current firmware upgrade

MODELS All

Access: R

Name	Type						
progress	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

firmware.component.green-controller-boot-v1.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS All

Access: R

Name	Type
requiredversion	string

firmware.component.green-controller-boot-v1.status

The status of the current firmware upgrade

MODELS All

Access: R

Name	Type					
status	enum					
	<table><tr><th>Values</th></tr><tr><td>"Inactive"</td></tr><tr><td>"Active"</td></tr><tr><td>"Error"</td></tr><tr><td>"Finished"</td></tr></table>	Values	"Inactive"	"Active"	"Error"	"Finished"
Values						
"Inactive"						
"Active"						
"Error"						
"Finished"						

firmware.component.green-controller-run-v1.actualversion

The version of the currently installed firmware.

MODELS All

Access: R

Name	Type
actualversion	string

firmware.component.green-controller-run-v1.displayname

The user-friendly name of the firmware component.

MODELS All

Access: R

Name	Type
displayname	string

firmware.component.green-controller-run-v1.error

The error message in case of an upgrade error.

MODELS All

Access: R

Name	Type
error	string

firmware.component.green-controller-run-v1.mismatch

Indicate whether there is a version mismatch

MODELS All

Access: R

Name	Type
mismatch	bool

firmware.component.green-controller-run-v1.progress

The progress of the current firmware upgrade

MODELS All

Access: R

Name	Type						
progress	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

firmware.component.green-controller-run-v1.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS All

Access: R

Name	Type
requiredversion	string

firmware.component.green-controller-run-v1.status

The status of the current firmware upgrade

MODELS All

Access: R

Name	Type
status	enum
Values	
"Inactive"	
"Active"	
"Error"	
"Finished"	

firmware.component.ld1-megmeet-psu.actualversion

The version of the currently installed firmware.

MODELS All

Access: R

Name	Type
actualversion	string

firmware.component.ld1-megmeet-psu.displayname

The user-friendly name of the firmware component.

MODELS All

Access: R

Name	Type
displayname	string

firmware.component.ld1-megmeet-psu.error

The error message in case of an upgrade error.

MODELS All

Access: R

Name	Type
error	string

firmware.component.ld1-megmeet-psu.mismatch

Indicate whether there is a version mismatch

MODELS

 All

Access: R

Name	Type
mismatch	bool

firmware.component.ld1-megmeet-psu.progress

The progress of the current firmware upgrade

MODELS

 All

Access: R

Name	Type
progress	{ }
Name	Type
base	int
hints	{ }
Name	Type

firmware.component.ld1-megmeet-psu.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS

 All

Access: R

Name	Type
requiredversion	string

firmware.component.ld1-megmeet-psu.status

The status of the current firmware upgrade

MODELS

 All

Access: R

Name	Type
status	enum
Values	
"Inactive"	
"Active"	
"Error"	
"Finished"	

firmware.component.ld2-megmeet-psu.actualversion

The version of the currently installed firmware.

MODELS

 All

Access: R

Name	Type
actualversion	string

firmware.component.ld2-megmeet-psu.displayName

The user-friendly name of the firmware component.

MODELS All

Access: R

Name	Type
displayname	string

firmware.component.ld2-megmeet-psu.error

The error message in case of an upgrade error.

MODELS All

Access: R

Name	Type
error	string

firmware.component.ld2-megmeet-psu.mismatch

Indicate whether there is a version mismatch

MODELS All

Access: R

Name	Type
mismatch	bool

firmware.component.ld2-megmeet-psu.progress

The progress of the current firmware upgrade

MODELS All

Access: R

Name	Type
progress	{ }
Name	Type
base	int
hints	{ }
Name	Type

firmware.component.ld2-megmeet-psu.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS

 All

Access: R

Name	Type
requiredversion	string

firmware.component.ld2-megmeet-psu.status

The status of the current firmware upgrade

MODELS

 All

Access: R

Name	Type
status	enum
Values	
"Inactive"	
"Active"	
"Error"	
"Finished"	

firmware.component.linux.actualversion

The version of the currently installed firmware.

MODELS

 All

Access: R

Name	Type
actualversion	string

firmware.component.linux.displayname

The user-friendly name of the firmware component.

MODELS

 All

Access: R

Name	Type
displayname	string

firmware.component.linux.error

The error message in case of an upgrade error.

MODELS

 All

Access: R

Name	Type
error	string

firmware.component.linux.mismatch

Indicate whether there is a version mismatch

MODELS All

Access: R

Name	Type
mismatch	bool

firmware.component.linux.progress

The progress of the current firmware upgrade

MODELS All

Access: R

Name	Type						
progress	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

firmware.component.linux.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS All

Access: R

Name	Type
requiredversion	string

firmware.component.linux.status

The status of the current firmware upgrade

MODELS All

Access: R

Name	Type
status	enum
	Values
	"Inactive"
	"Active"
	"Error"
	"Finished"

firmware.component.mysticum-image-processing-3-mk22-boot.actualversion

The version of the currently installed firmware.

MODELS

All

Access: R

Name	Type
actualversion	string

firmware.component.mysticum-image-processing-3-mk22-boot.displayname

The user-friendly name of the firmware component.

MODELS

All

Access: R

Name	Type
displayname	string

firmware.component.mysticum-image-processing-3-mk22-boot.error

The error message in case of an upgrade error.

MODELS

All

Access: R

Name	Type
error	string

firmware.component.mysticum-image-processing-3-mk22-boot.mismatch

Indicate whether there is a version mismatch

MODELS

All

Access: R

Name	Type
mismatch	bool

firmware.component.mysticum-image-processing-3-mk22-boot.progress

The progress of the current firmware upgrade

MODELS

All

Access: R

Name	Type
progress	{ }
Name	Type
base	int
hints	{ }
Name	Type

firmware.component.mysticum-image-processing-3-mk22-boot.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS All

Access: R

Name	Type
requiredversion	string

firmware.component.mysticum-image-processing-3-mk22-boot.status

The status of the current firmware upgrade

MODELS All

Access: R

Name	Type
status	enum
Values	
"Inactive"	
"Active"	
"Error"	
"Finished"	

firmware.component.mysticum-image-processing-3-mk22-fpga.actualversion

The version of the currently installed firmware.

MODELS All

Access: R

Name	Type
actualversion	string

firmware.component.mysticum-image-processing-3-mk22-fpga.displayname

The user-friendly name of the firmware component.

MODELS All

Access: R

Name	Type
displayname	string

firmware.component.mysticum-image-processing-3-mk22-fpga.error

The error message in case of an upgrade error.

MODELS All

Access: R

Name	Type
error	string

firmware.component.mysticum-image-processing-3-mk22-fpga.mismatch

Indicate whether there is a version mismatch

MODELS All

Access: R

Name	Type
mismatch	bool

firmware.component.mysticum-image-processing-3-mk22-fpga.progress

The progress of the current firmware upgrade

MODELS All

Access: R

Name	Type						
progress	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

firmware.component.mysticum-image-processing-3-mk22-fpga.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS All

Access: R

Name	Type
requiredversion	string

firmware.component.mysticum-image-processing-3-mk22-fpga.status

The status of the current firmware upgrade

MODELS All

Access: R

Name	Type				
status	enum				
	<table><tr><th>Values</th></tr><tr><td>"Inactive"</td></tr><tr><td>"Active"</td></tr><tr><td>"Error"</td></tr></table>	Values	"Inactive"	"Active"	"Error"
Values					
"Inactive"					
"Active"					
"Error"					

"Finished"

firmware.component.mysticum-image-processing-3-mk22-run.actualversion

The version of the currently installed firmware.

MODELS All

Access: R

Name	Type
actualversion	string

firmware.component.mysticum-image-processing-3-mk22-run.displayname

The user-friendly name of the firmware component.

MODELS All

Access: R

Name	Type
displayname	string

firmware.component.mysticum-image-processing-3-mk22-run.error

The error message in case of an upgrade error.

MODELS All

Access: R

Name	Type
error	string

firmware.component.mysticum-image-processing-3-mk22-run.mismatch

Indicate whether there is a version mismatch

MODELS All

Access: R

Name	Type
mismatch	bool

firmware.component.mysticum-image-processing-3-mk22-run.progress

The progress of the current firmware upgrade

MODELS All

Access: R

Name	Type		
progress	{ }		
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type
Name	Type		

```
base    int
hints   { }
```

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

firmware.component.mysticum-image-processing-3-mk22-run.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS

 All

Access: R

Name	Type
requiredversion	string

firmware.component.mysticum-image-processing-3-mk22-run.status

The status of the current firmware upgrade

MODELS

 All

Access: R

Name	Type
status	enum
Values	
"Inactive"	
"Active"	
"Error"	
"Finished"	

firmware.component.redstar-fan-motor-control-board-fpga.actualversion

The version of the currently installed firmware.

MODELS

 All

Access: R

Name	Type
actualversion	string

firmware.component.redstar-fan-motor-control-board-fpga.displayname

The user-friendly name of the firmware component.

MODELS

 All

Access: R

Name	Type
displayname	string

firmware.component.redstar-fan-motor-control-board-fpga.error

The error message in case of an upgrade error.

MODELS All

Access: R

Name	Type
error	string

firmware.component.redstar-fan-motor-control-board-fpga.mismatch

Indicate whether there is a version mismatch

MODELS All

Access: R

Name	Type
mismatch	bool

firmware.component.redstar-fan-motor-control-board-fpga.progress

The progress of the current firmware upgrade

MODELS All

Access: R

Name	Type						
progress	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

firmware.component.redstar-fan-motor-control-board-fpga.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS All

Access: R

Name	Type
requiredversion	string

firmware.component.redstar-fan-motor-control-board-fpga.status

The status of the current firmware upgrade

MODELS All

Access: R

Name	Type
status	enum

Values
"Inactive"
"Active"
"Error"
"Finished"

firmware.component.redstar-fan-motor-control-boot.actualversion

The version of the currently installed firmware.

MODELS All

Access: R

Name	Type
actualversion	string

firmware.component.redstar-fan-motor-control-boot.displayname

The user-friendly name of the firmware component.

MODELS All

Access: R

Name	Type
displayname	string

firmware.component.redstar-fan-motor-control-boot.error

The error message in case of an upgrade error.

MODELS All

Access: R

Name	Type
error	string

firmware.component.redstar-fan-motor-control-boot.mismatch

Indicate whether there is a version mismatch

MODELS All

Access: R

Name	Type
mismatch	bool

firmware.component.redstar-fan-motor-control-boot.progress

The progress of the current firmware upgrade

MODELS All

Access: R

Name	Type						
progress	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

firmware.component.redstar-fan-motor-control-boot.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS All

Access: R

Name	Type
requiredversion	string

firmware.component.redstar-fan-motor-control-boot.status

The status of the current firmware upgrade

MODELS All

Access: R

Name	Type					
status	enum					
	<table><tr><th>Values</th></tr><tr><td>"Inactive"</td></tr><tr><td>"Active"</td></tr><tr><td>"Error"</td></tr><tr><td>"Finished"</td></tr></table>	Values	"Inactive"	"Active"	"Error"	"Finished"
Values						
"Inactive"						
"Active"						
"Error"						
"Finished"						

firmware.component.redstar-fan-motor-control-run.actualversion

The version of the currently installed firmware.

MODELS All

Access: R

Name	Type
actualversion	string

firmware.component.redstar-fan-motor-control-run.displayname

The user-friendly name of the firmware component.

MODELS All

Access: R

Name	Type
displayname	string

firmware.component.redstar-fan-motor-control-run.error

The error message in case of an upgrade error.

MODELS All

Access: R

Name	Type
error	string

firmware.component.redstar-fan-motor-control-run.mismatch

Indicate whether there is a version mismatch

MODELS All

Access: R

Name	Type
mismatch	bool

firmware.component.redstar-fan-motor-control-run.progress

The progress of the current firmware upgrade

MODELS All

Access: R

Name	Type						
progress	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

firmware.component.redstar-fan-motor-control-run.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS All

Access: R

Name	Type
requiredversion	string

firmware.component.redstar-fan-motor-control-run.status

The status of the current firmware upgrade

MODELS All

Access: R

Name	Type
status	enum
Values	
"Inactive"	
"Active"	
"Error"	
"Finished"	

firmware.component.status

The status of the current component firmware upgrade

MODELS All

Access: R

Name	Type
status	enum
Values	
"Inactive"	
"Active"	
"Error"	
"Finished"	

firmware.component.u-boot.actualversion

The version of the currently installed firmware.

MODELS All

Access: R

Name	Type
actualversion	string

firmware.component.u-boot.displayname

The user-friendly name of the firmware component.

MODELS All

Access: R

Name	Type
displayname	string

firmware.component.u-boot.error

The error message in case of an upgrade error.

MODELS All

Access: R

Name	Type
error	string

firmware.component.u-boot.mismatch

Indicate whether there is a version mismatch

MODELS All

Access: R

Name	Type
mismatch	bool

firmware.component.u-boot.progress

The progress of the current firmware upgrade

MODELS All

Access: R

Name	Type						
progress	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

firmware.component.u-boot.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS All

Access: R

Name	Type
requiredversion	string

firmware.component.u-boot.status

The status of the current firmware upgrade

MODELS All

Access: R

Name	Type			
status	enum			
	<table><tr><th>Values</th></tr><tr><td>"Inactive"</td></tr><tr><td>"Active"</td></tr></table>	Values	"Inactive"	"Active"
Values				
"Inactive"				
"Active"				

"Error"

"Finished"

firmware.component.zynq-fpga.actualversion

The version of the currently installed firmware.

MODELS All

Access: R

Name	Type
actualversion	string

firmware.component.zynq-fpga.displayname

The user-friendly name of the firmware component.

MODELS All

Access: R

Name	Type
displayname	string

firmware.component.zynq-fpga.error

The error message in case of an upgrade error.

MODELS All

Access: R

Name	Type
error	string

firmware.component.zynq-fpga.mismatch

Indicate whether there is a version mismatch

MODELS All

Access: R

Name	Type
mismatch	bool

firmware.component.zynq-fpga.progress

The progress of the current firmware upgrade

MODELS All

Access: R

Name	Type
progress	{ }

Name	Type
base	int
hints	{ }

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

firmware.component.zynq-fpga.requiredversion

The firmware version this component is expected to have by the current firmware package.

MODELS All

Access: R

Name	Type
requiredversion	string

firmware.component.zynq-fpga.status

The status of the current firmware upgrade

MODELS All

Access: R

Name	Type
status	enum
Values	
"Inactive"	
"Active"	
"Error"	
"Finished"	

firmware.error

The error of the last firmware upgrade (only set if status is error)

MODELS All

Access: R

Name	Type
error	string

firmware.eula.requiretermagreement

Indiates whether or not the user is required to accept the term agreement.

MODELS All

Access: R

Name	Type
requiretermagreement	bool

firmware.eula.termagreement

The user agreement with the firmware EULA.

MODELS

 All

Access: RW

Name	Type
termagreement	bool

firmware.firmwareversion

The version of the currently installed firmware.

MODELS

 All

Access: R

Name	Type
firmwareversion	string

firmware.progress

The progress of the current firmware upgrade

MODELS

 All

Access: R

Name	Type						
progress	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

firmware.status

The status of the current firmware upgrade

MODELS

 All

Access: R

Name	Type					
status	enum					
	<table><tr><th>Values</th></tr><tr><td>"Inactive"</td></tr><tr><td>"Active"</td></tr><tr><td>"Error"</td></tr><tr><td>"Finished"</td></tr></table>	Values	"Inactive"	"Active"	"Error"	"Finished"
Values						
"Inactive"						
"Active"						
"Error"						
"Finished"						

gpio.gpo.trigger1.description

Description of gpo

MODELS

 All

Access: R

Name	Type
description	string

gpio.gpo.trigger1.mode

Mode of output, level or pulsetrain

MODELS

 All

Access: RW

Name	Type
mode	enum
Values	
"Level"	
"PulseTrain"	

gpio.gpo.trigger1.name

Name of gpo

MODELS

 All

Access: R

Name	Type
name	string

gpio.gpo.trigger1.reserved

Is GPIO reserved

MODELS

 All

Access: R

Name	Type
reserved	bool

gpio.gpo.trigger1.value

Value of output

MODELS

 All

Access: RW

Name	Type
value	bool

gpio.gpo.trigger2.description

Description of gpo

MODELS

 All

Access: R

Name	Type
description	string

gpio.gpo.trigger2.mode

Mode of output, level or pulsetrain

MODELS

 All

Access: RW

Name	Type
mode	enum
Values	
"Level"	
"PulseTrain"	

gpio.gpo.trigger2.name

Name of gpo

MODELS

 All

Access: R

Name	Type
name	string

gpio.gpo.trigger2.reserved

Is GPIO reserved

MODELS

 All

Access: R

Name	Type
reserved	bool

gpio.gpo.trigger2.value

Value of output

MODELS

 All

Access: RW

Name	Type
value	bool

gsm.data.status

The current status of the mobile connection.

MODELS All

Access: R

Name	Type
status	enum
Values	
"FAILED"	
"UNKNOWN"	
"INITIALIZING"	
"LOCKED"	
"DISABLED"	
"DISABLING"	
"ENABLING"	
"ENABLED"	
"SEARCHING"	
"REGISTERED"	
"DISCONNECTING"	
"CONNECTING"	
"CONNECTED"	

gsm.iccid

The ICCID that identifies the SIM.

MODELS All

Access: R

Name	Type
iccid	string

gsm.imei

The IMEI that identifies the GSM module.

MODELS All

Access: R

Name	Type
imei	string

gsm.signalstrength

The signal strength in percent.

MODELS All

Access: R

Name	Type
signalstrength	{ }
Name	Type
base	int
hints	{ }
Name	Type

illumination.clo.allowed

Indicates which features are preventing CLO from being enabled

MODELS All

Access: R

Name	Type
allowed	{ }
Name	Type
allowed	bool
reasons	[string]

illumination.clo.availability

Shows the current availability.

MODELS All

Access: R

Name	Type
availability	enum
Values	
"Available"	
"SensorUnavailable"	
"AutoDimming"	
"PendingWarmup"	
"PendingShutdown"	
"Unavailable"	
"Unknown"	

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	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

illumination.clo.setpoint

The target luminosity of the light source

MODELS

 All

Access: RW

Name	Type						
setpoint	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

illumination.clo.state

State of the CLO

MODELS

 All

Access: R

Name	Type
state	enum
Values	
"Idle"	
"Ok"	
"TooDim"	
"TooBright"	
"Pending"	
"Error"	

illumination.nosignal.dimming.enable

Enable/disable NoSignal dimming of the power

MODELS

 All

Access: RW

Name	Type
enable	bool

illumination.nosignal.dimming.power

The power limitation (in percent) that the NoSignal dimming will apply

MODELS

 All

Access: R

Name	Type						
power	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						

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

illumination.sources.laser.actualpower

actual power in percent with limits

MODELS

 All

Access: R

Name	Type						
actualpower	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						

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

illumination.sources.laser.controlledbyclo

Describes whether CLO is controlling this source.

MODELS

 All

Access: R

Name	Type
controlledbyclo	bool

illumination.sources.laser.ispowerlimited

Whether power is currently limited.

MODELS

 All

Access: R

Name	Type
ispowerlimited	bool

illumination.sources.laser.maxpower

maximum power in percent

MODELS All

Access: R

Name	Type						
maxpower	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						

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

illumination.sources.laser.minpower

minimum power in percent

MODELS All

Access: R

Name	Type						
minpower	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						

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

illumination.sources.laser.power

target power in percent

MODELS All

Access: RW

Name	Type						
power	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						

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

illumination.sources.laser.powerlimitreason

Contains the reason of the authority that limited the power.

MODELS All

Access: R

Name	Type
powerlimitreason	string

illumination.sources.laser.powerprecision

Power precision this source takes into account, expressed as number of digits to the right of the decimal point.

MODELS All

Access: R

Name	Type						
powerprecision	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

illumination.sources.laser.status

Description not provided

MODELS All

Access: R

Name	Type								
status	enum								
	<table><tr><th>Values</th></tr><tr><td>"Off"</td></tr><tr><td>"Ignite"</td></tr><tr><td>"WarmingUp"</td></tr><tr><td>"On"</td></tr><tr><td>"Extinguish"</td></tr><tr><td>"Cooling"</td></tr><tr><td>"Malfunctioning"</td></tr></table>	Values	"Off"	"Ignite"	"WarmingUp"	"On"	"Extinguish"	"Cooling"	"Malfunctioning"
Values									
"Off"									
"Ignite"									
"WarmingUp"									
"On"									
"Extinguish"									
"Cooling"									
"Malfunctioning"									

illumination.sources.laser.timcuring.dmd.cureavailable

Description not provided

MODELS All

Access: R

Name	Type
cureavailable	bool

illumination.sources.laser.timcuring.dmd.state

State of the current or past curing execution expressed in the format of a date, a status and a list of failed plates

MODELS

All

Access: R

Name	Type
state	{ }

Name	Type
date	string
status	enum

Values
"Inactive"
"Preparing"
"Active"
"Cooling"
"Success"
"Failure"

failures	[{ }]
	{ }

Name	Type
base	int
hints	{ }

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

illumination.state

The state of light

MODELS

All

Access: R

Name	Type			
state	enum			
	<table><tr><th>Values</th></tr><tr><td>"On"</td></tr><tr><td>"Off"</td></tr></table>	Values	"On"	"Off"
Values				
"On"				
"Off"				

image.brightness

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

MODELS

All

Access: RW

Name	Type			
brightness	float			
	<table><tr><th>Constraints</th></tr><tr><td>Minimum</td><td>-1</td></tr></table>	Constraints	Minimum	-1
Constraints				
Minimum	-1			

Maximum	1
Step size	1
Precision	0.01

image.brilliantcolor.mode

The current BrilliantColor mode.

MODELS All

Access: RW

Name	Type
mode	string

image.color.p7.custom.bluegain

Desired blue gain value

MODELS All

Access: RW

Name	Type						
bluegain	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

image.color.p7.custom.bluelum

Desired blue luminanace

MODELS All

Access: RW

Name	Type										
bluelum	float										
<table><tr><th colspan="2">Constraints</th></tr><tr><td>Minimum</td><td>0.0001</td></tr><tr><td>Maximum</td><td>1</td></tr><tr><td>Step size</td><td>0.0001</td></tr><tr><td>Precision</td><td>1</td></tr></table>		Constraints		Minimum	0.0001	Maximum	1	Step size	0.0001	Precision	1
Constraints											
Minimum	0.0001										
Maximum	1										
Step size	0.0001										
Precision	1										

image.color.p7.custom.bluex

Desired blue x in xy-coordinates

MODELS All

Access: RW

Name	Type
bluex	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.bluey

Desired blue y in xy-coordinates

MODELS

All

Access: RW

Name	Type
bluey	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

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

MODELS

All

Access: R

Name	Type
cmywritable	bool

image.color.p7.custom.cyangain

Desired cyan gain value

MODELS

All

Access: RW

Name	Type
cyangain	{ }
Name	Type
base	float
hints	{ }
Name	Type

image.color.p7.custom.cyanlum

Desired cyan luminanace

MODELS All

Access: RW

Name	Type
cyanlum	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.cyanx

Desired cyan x in xy-coordinates

MODELS All

Access: RW

Name	Type
cyanx	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.cyany

Desired cyan y in xy-coordinates

MODELS All

Access: RW

Name	Type
cyany	float
Constraints	
Minimum	0.0001

Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.gainsavailable

true when gains are available

MODELS

 All

Access: R

Name	Type
gainsavailable	bool

image.color.p7.custom.greengain

Desired green gain value

MODELS

 All

Access: RW

Name	Type
greengain	{ }
Name	Type
base	float
hints	{ }
Name	Type

image.color.p7.custom.greenlum

Desired green luminanace

MODELS

 All

Access: RW

Name	Type
greenlum	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.greenx

Desired green x in xy-coordinates

MODELS

 All

Access: RW

Name	Type
greenx	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.greeny

Desired green y in xy-coordinates

MODELS All

Access: RW

Name	Type
greeny	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.luminancesavailable

true if luminances are available

MODELS All

Access: R

Name	Type
luminancesavailable	bool

image.color.p7.custom.magentagain

Desired magenta gain value

MODELS All

Access: RW

Name	Type
magentagain	{ }
Name Type base float hints { }	
Name Type	

image.color.p7.custom.magentalum

Desired magenta luminanace

MODELS

 All

Access: RW

Name	Type
magentalum	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.magentax

Desired magenta x in xy-coordinates

MODELS

 All

Access: RW

Name	Type
magentax	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.magentay

Desired magenta y in xy-coordinates

MODELS

 All

Access: RW

Name	Type
magentay	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.mode

Description not provided

MODELS

 All

Access: RW

Name	Type
mode	string

image.color.p7.custom.modes

Description not provided

MODELS All

Access: R

Name	Type											
modes	[{ }]											
	{ }											
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>group</td><td>enum</td></tr><tr><td></td><td><table><tr><th>Values</th></tr><tr><td>"Custom"</td></tr><tr><td>"Preset"</td></tr></table></td></tr><tr><td>modes</td><td>[string]</td></tr></table>	Name	Type	group	enum		<table><tr><th>Values</th></tr><tr><td>"Custom"</td></tr><tr><td>"Preset"</td></tr></table>	Values	"Custom"	"Preset"	modes	[string]
Name	Type											
group	enum											
	<table><tr><th>Values</th></tr><tr><td>"Custom"</td></tr><tr><td>"Preset"</td></tr></table>	Values	"Custom"	"Preset"								
Values												
"Custom"												
"Preset"												
modes	[string]											

image.color.p7.custom.redgain

Desired red gain value

MODELS All

Access: RW

Name	Type						
redgain	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

image.color.p7.custom.redlum

Desired red luminanace

MODELS All

Access: RW

Name	Type					
redlum	float					
	<table><tr><th>Constraints</th></tr><tr><td>Minimum</td><td>0.0001</td></tr><tr><td>Maximum</td><td>1</td></tr></table>	Constraints	Minimum	0.0001	Maximum	1
Constraints						
Minimum	0.0001					
Maximum	1					

Step size	0.0001
Precision	1

image.color.p7.custom.redx

Desired red x in xy-coordinates

MODELS All

Access: RW

Name	Type
redx	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.redy

Desired red y in xy-coordinates

MODELS All

Access: RW

Name	Type
redy	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.rgbcmYGainsAvailable

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

MODELS All

Access: R

Name	Type
rgbcmYGainsAvailable	bool

image.color.p7.custom.rgbwritable

true if RGB are Writable (not in WHITE mode)

MODELS All

Access: R

Name	Type
rgbwritable	bool

image.color.p7.custom.target

Target color coordinates for the current preset

MODELS All

Access: R

Name	Type																																																																																																				
target	{ }																																																																																																				
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>red</td><td>{ }</td></tr></table>	Name	Type	red	{ }																																																																																																
Name	Type																																																																																																				
red	{ }																																																																																																				
	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr><tr><td>green</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr><tr><td>blue</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table></td></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr><tr><td>green</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr><tr><td>blue</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	y	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	green	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr><tr><td>blue</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr><tr><td>blue</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	y	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	blue	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }																																																																																																
Name	Type																																																																																																				
x	{ }																																																																																																				
	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr><tr><td>green</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr><tr><td>blue</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	y	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	green	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr><tr><td>blue</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr><tr><td>blue</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	y	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	blue	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type								
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																																																																																														
Name	Type																																																																																																				
base	float																																																																																																				
hints	{ }																																																																																																				
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																																																																		
Name	Type																																																																																																				
y	{ }																																																																																																				
	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																																																								
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																																																																																														
Name	Type																																																																																																				
base	float																																																																																																				
hints	{ }																																																																																																				
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																																																																		
Name	Type																																																																																																				
green	{ }																																																																																																				
	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr><tr><td>blue</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr><tr><td>blue</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	y	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	blue	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																								
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }																																																																																																
Name	Type																																																																																																				
x	{ }																																																																																																				
	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr><tr><td>blue</td><td>{ }</td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	y	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	blue	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																																																																																														
Name	Type																																																																																																				
base	float																																																																																																				
hints	{ }																																																																																																				
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																																																																		
Name	Type																																																																																																				
y	{ }																																																																																																				
	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																																																								
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																																																																																														
Name	Type																																																																																																				
base	float																																																																																																				
hints	{ }																																																																																																				
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																																																																		
Name	Type																																																																																																				
blue	{ }																																																																																																				
	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																																																
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }																																																																																																
Name	Type																																																																																																				
x	{ }																																																																																																				
	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																																																								
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																																																																																														
Name	Type																																																																																																				
base	float																																																																																																				
hints	{ }																																																																																																				
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																																																																		
Name	Type																																																																																																				

y	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						
white	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>		Name	Type	x	{ }		
Name	Type						
x	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						
y	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

image.color.p7.custom.whitegain

Desired white gain value

MODELS All

Access: RW

Name	Type						
whitegain	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

image.color.p7.custom.whitegainavailable

true when white gain is available

MODELS All

Access: R

Name	Type
whitegainavailable	bool

image.color.p7.custom.whitelum

Desired white luminanace

MODELS All

Access: RW

Name	Type
whitelum	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.whitemode

Description not provided

MODELS

All

Access: RW

Name	Type
whitemode	enum
Values	
"Coordinates"	
"Temperature"	

image.color.p7.custom.whitetemperature

Desired white point temperature

MODELS

All

Access: RW

Name	Type
whitetemperature	int
Constraints	
Minimum	3200
Maximum	13000
Step size	100
Precision	1

image.color.p7.custom.whitetemperatureavailable

true if White temperature is available

MODELS

All

Access: R

Name	Type
whitetemperatureavailable	bool

image.color.p7.custom.whitewritable

true if White is Writable

MODELS

 All

Access: R

Name	Type
whitewritable	bool

image.color.p7.custom.whitex

Desired white x in xy-coordinates

MODELS

 All

Access: RW

Name	Type
whitex	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.whitey

Desired white y in xy-coordinates

MODELS

 All

Access: RW

Name	Type
whitey	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.yellowgain

Desired yellow gain value

MODELS

 All

Access: RW

Name	Type
yellowgain	{ }
NameType	
base	float

hints { }

Name Type

image.color.p7.custom.yellowlum

Desired yellow luminanace

MODELS All

Access: RW

Name	Type
yellowlum	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.yellowx

Desired yellow x in xy-coordinates

MODELS All

Access: RW

Name	Type
yellowx	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.custom.yellowy

Desired yellow y in xy-coordinates

MODELS All

Access: RW

Name	Type
yellowy	float
Constraints	
Minimum	0.0001
Maximum	1
Step size	0.0001
Precision	1

image.color.p7.native.c1

Native C1 x in xy-coordinates

MODELS All

Access: R

Name	Type																																												
c1	{ }																																												
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>lum</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>	Name	Type	x	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	y	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	lum	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type
Name	Type																																												
x	{ }																																												
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																																						
Name	Type																																												
base	float																																												
hints	{ }																																												
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																										
Name	Type																																												
y	{ }																																												
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																																						
Name	Type																																												
base	float																																												
hints	{ }																																												
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																										
Name	Type																																												
lum	{ }																																												
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																																						
Name	Type																																												
base	float																																												
hints	{ }																																												
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																										
Name	Type																																												

image.color.p7.native.c1available

Description not provided

MODELS All

Access: R

Name	Type
c1available	bool

image.color.p7.native.c2

Native C2 x in xy-coordinates

MODELS All

Access: R

Name	Type				
c2	{ }				
<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>		Name	Type	x	{ }
Name	Type				
x	{ }				

		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type							
base	float							
hints	{ }							
y	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type							
		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type							
base	float							
hints	{ }							
lum	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type							
		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type							
base	float							
hints	{ }							
		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type							

image.color.p7.native.c2available

Description not provided

MODELS All

Access: R

Name	Type
c2available	bool

image.color.p7.native.list

list available native sets

MODELS All

Access: R

Name	Type
list	[string]

image.color.p7.native.normal.c1

Native C1 x in xy-coordinates

MODELS All

Access: R

Name	Type
c1	{ }
Name	Type
x	{ }
Name	Type
base	float

		hints	{ }			
				<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type
Name	Type					
y	{ }					
		Name	Type			
		base	float			
		hints	{ }			
				<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type
Name	Type					
lum	{ }					
		Name	Type			
		base	float			
		hints	{ }			
				<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type
Name	Type					

image.color.p7.native.normal.c1available

Description not provided

MODELS All

Access: R

Name	Type
c1available	bool

image.color.p7.native.normal.c2

Native C2 x in xy-coordinates

MODELS All

Access: R

Name	Type
c2	{ }
Name	Type
x	{ }
Name	Type
base	float
hints	{ }
Name	Type
y	{ }
Name	Type
base	float
hints	{ }
Name	Type
lum	{ }
Name	Type
base	float

hints { }

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

image.color.p7.native.normal.c2available

Description not provided

MODELS All

Access: R

Name	Type
c2available	bool

image.color.p7.native.normal.rgbw

Native red x in xy-coordinates

MODELS All

Access: R

Name	Type																																																																						
rgbw	{ }																																																																						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>red</td><td>{ }</td></tr></table>		Name	Type	red	{ }																																																																		
Name	Type																																																																						
red	{ }																																																																						
<table><tr><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr></table></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>lum</td><td>{ }</td></tr></table></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>green</td><td>{ }</td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	<table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr></table>	Name	Type	y	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	<table><tr><th>Name</th><th>Type</th></tr><tr><td>lum</td><td>{ }</td></tr></table>	Name	Type	lum	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	green	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>		Name	Type	x	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																																																												
Name	Type																																																																						
x	{ }																																																																						
Name	Type																																																																						
base	float																																																																						
hints	{ }																																																																						
<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																																		
Name	Type																																																																						
Name	Type																																																																						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr></table>	Name	Type	y	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																																																												
Name	Type																																																																						
y	{ }																																																																						
Name	Type																																																																						
base	float																																																																						
hints	{ }																																																																						
<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																																		
Name	Type																																																																						
Name	Type																																																																						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>lum</td><td>{ }</td></tr></table>	Name	Type	lum	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																																																												
Name	Type																																																																						
lum	{ }																																																																						
Name	Type																																																																						
base	float																																																																						
hints	{ }																																																																						
<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																																		
Name	Type																																																																						
Name	Type																																																																						
green	{ }																																																																						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>		Name	Type	x	{ }																																																																		
Name	Type																																																																						
x	{ }																																																																						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }																																																																
Name	Type																																																																						
base	float																																																																						
hints	{ }																																																																						

		Name	Type	
y	{ }			
		Name	Type	
		base	float	
		hints	{ }	
lum	{ }			
		Name	Type	
		base	float	
		hints	{ }	
blue	{ }			
		Name	Type	
		x	{ }	
		Name	Type	
		base	float	
		hints	{ }	
y	{ }			
		Name	Type	
		base	float	
		hints	{ }	
lum	{ }			
		Name	Type	
		base	float	
		hints	{ }	
white	{ }			
		Name	Type	
		x	{ }	
		Name	Type	
		base	float	
		hints	{ }	
y	{ }			
		Name	Type	
		base	float	
		hints	{ }	
lum	{ }			
		Name	Type	

base	float
hints	{ }

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

image.color.p7.native.normal.whiteavailable

Description not provided

MODELS All

Access: R

Name	Type
whiteavailable	bool

image.color.p7.native.rgbw

Native red x in xy-coordinates

MODELS All

Access: R

Name	Type																																															
rgbw	{ }																																															
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>red</td><td>{ }</td></tr></table>	Name	Type	red	{ }																																											
Name	Type																																															
red	{ }																																															
	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td>lum</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td>green</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }	y	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }	lum	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }	green	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr></table>	Name	Type	base	float
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																																				
Name	Type																																															
x	{ }																																															
Name	Type																																															
base	float																																															
hints	{ }																																															
y	{ }																																															
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																																									
Name	Type																																															
base	float																																															
hints	{ }																																															
lum	{ }																																															
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																																									
Name	Type																																															
base	float																																															
hints	{ }																																															
green	{ }																																															
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }																																											
Name	Type																																															
x	{ }																																															
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr></table>	Name	Type	base	float																																											
Name	Type																																															
base	float																																															

		hints	{ }
		Name Type	
y	{ }		
		Name Type	
		base	float
		hints	{ }
		Name Type	
lum	{ }		
		Name Type	
		base	float
		hints	{ }
		Name Type	
blue	{ }		
		Name Type	
		x	{ }
		Name Type	
		base	float
		hints	{ }
		Name Type	
y	{ }		
		Name Type	
		base	float
		hints	{ }
		Name Type	
lum	{ }		
		Name Type	
		base	float
		hints	{ }
		Name Type	
white	{ }		
		Name Type	
		x	{ }
		Name Type	
		base	float
		hints	{ }
		Name Type	
y	{ }		
		Name Type	
		base	float
		hints	{ }
		Name Type	
lum	{ }		

Name	Type
base	float
hints	{ }

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

image.color.p7.native.whiteavailable

Description not provided

MODELS All

Access: R

Name	Type
whiteavailable	bool

image.color.rgbmode.rgbmode

RGB Mode

MODELS All

Access: RW

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

image.connector.displayport.capabilities

Capabilities.

MODELS All

Access: R

Name	Type
capabilities	[enum]
Values	
"HDCP"	
"HDCP2"	

image.connector.displayport.colorprimaries

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

MODELS

 All

Access: RW

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

image.connector.displayport.colorprimariesavailable

true if Color Primaries is available

MODELS

 All

Access: R

Name	Type
colorprimariesavailable	bool

image.connector.displayport.colorspace

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

MODELS

 All

Access: RW

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

image.connector.displayport.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

 All

Access: R

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

detectedsignal { }									
<table><tr><th>Name</th><th>Type</th></tr><tr><td>active</td><td>bool</td></tr><tr><td>name</td><td>string</td></tr><tr><td>vertical_total</td><td>{ }</td></tr></table>		Name	Type	active	bool	name	string	vertical_total	{ }
Name	Type								
active	bool								
name	string								
vertical_total	{ }								
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }		
Name	Type								
base	int								
hints	{ }								
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type						
Name	Type								
<table><tr><td>horizontal_total</td><td>{ }</td></tr></table>		horizontal_total	{ }						
horizontal_total	{ }								
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }		
Name	Type								
base	int								
hints	{ }								
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type						
Name	Type								
<table><tr><td>vertical_resolution</td><td>{ }</td></tr></table>		vertical_resolution	{ }						
vertical_resolution	{ }								
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }		
Name	Type								
base	int								
hints	{ }								
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type						
Name	Type								
<table><tr><td>horizontal_resolution</td><td>{ }</td></tr></table>		horizontal_resolution	{ }						
horizontal_resolution	{ }								
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }		
Name	Type								
base	int								
hints	{ }								
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type						
Name	Type								
<table><tr><td>vertical_sync_width</td><td>{ }</td></tr></table>		vertical_sync_width	{ }						
vertical_sync_width	{ }								
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }		
Name	Type								
base	int								
hints	{ }								
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type						
Name	Type								
<table><tr><td>vertical_front_porch</td><td>{ }</td></tr></table>		vertical_front_porch	{ }						
vertical_front_porch	{ }								
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }		
Name	Type								
base	int								
hints	{ }								
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type						
Name	Type								
<table><tr><td>vertical_back_porch</td><td>{ }</td></tr></table>		vertical_back_porch	{ }						
vertical_back_porch	{ }								
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }		
Name	Type								
base	int								
hints	{ }								
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type						
Name	Type								
<table><tr><td>horizontal_sync_width</td><td>{ }</td></tr></table>		horizontal_sync_width	{ }						
horizontal_sync_width	{ }								
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr></table>		Name	Type	base	int				
Name	Type								
base	int								

horizontal_front_porch	{ }	hints	{ }		
				Name	Type
		Name	Type		
		base	int		
		hints	{ }		
horizontal_back_porch	{ }			Name	Type
		Name	Type		
		base	int		
		hints	{ }		
horizontal_frequency	{ }			Name	Type
		Name	Type		
		base	float		
		hints	{ }		
vertical_frequency	{ }			Name	Type
		Name	Type		
		base	float		
		hints	{ }		
pixel_rate	{ }			Name	Type
		Name	Type		
		base	int		
		hints	{ }		
scan	enum			Name	Type
		Values			
bits_per_component	{ }				
		Name	Type		
		base	int		
		hints	{ }		
color_space	enum			Name	Type
		Values			

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" "SIM1" "SMPTE_ST2084" "DICOM_10" "DICOM_60" "DICOM_180" "DICOM_250" "DICOM_300" "DICOM_400"						
color_primaries	enum	Values "REC709" "REC2020" "DCI-P3-D65" "DCI-P3-Theater"						
mastering_luminance	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type							
base	float							
hints	{ }							
max_c11	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type							
base	float							
hints	{ }							
max_fall	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type							
base	float							
hints	{ }							

		Name	Type
content_aspect_ratio	enum	Values	
		"5:4"	
		"4:3"	
		"16:10"	
		"16:9"	
		"1.85:1"	
		"1.88:1"	
		"1.9:1"	
		"2.20:1"	
		"2.00:1"	
		"2.35:1"	
		"2.37:1"	
		"2.39:1"	
		"Unknown"	
content_type	enum	Values	
		"Graphics"	
		"Photo"	
		"Cinema"	
		"Game"	
		"Testpattern"	
is_stereo	bool	"Unknown"	
stereo_mode	enum	Values	
		"None"	
		"Sequential"	
		"FramePacked"	
		"TopBottom"	
		"SideBySide"	

image.connector.displayport.edid.selected

Selected EDID for connector

MODELS

All

Access: RW

Name	Type
selected	string

image.connector.displayport.name

Name of the source

MODELS

 All

Access: R

Name	Type
name	string

image.connector.displayport.signalrange

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

MODELS

 All

Access: RW

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

image.connector.hdmi.capabilities

Capabilities.

MODELS

 All

Access: R

Name	Type
capabilities	[enum]
Values	
"HDCP"	
"HDCP2"	

image.connector.hdmi.colorprimaries

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

MODELS

 All

Access: RW

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

"DCI-P3-Theater"

image.connector.hdmi.colorprimariesavailable

true if Color Primaries is available

MODELS All

Access: R

Name	Type
colorprimariesavailable	bool

image.connector.hdmi.colorspace

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

MODELS All

Access: RW

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

image.connector.hdmi.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 All

Access: R

Name	Type								
detectedsignal	{ }								
<table><tr><th>Name</th><th>Type</th></tr><tr><td>active</td><td>bool</td></tr><tr><td>name</td><td>string</td></tr><tr><td>vertical_total</td><td>{ }</td></tr></table>		Name	Type	active	bool	name	string	vertical_total	{ }
Name	Type								
active	bool								
name	string								
vertical_total	{ }								
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }		
Name	Type								
base	int								
hints	{ }								
<table><tr><th>Name</th><th>Type</th></tr><tr><td>horizontal_total</td><td>{ }</td></tr></table>		Name	Type	horizontal_total	{ }				
Name	Type								
horizontal_total	{ }								

vertical_resolution	{ }	base	int	
		hints	{ }	
		Name	Type	
		base	int	
horizontal_resolution	{ }	base	int	
		hints	{ }	
		Name	Type	
		base	int	
vertical_sync_width	{ }	base	int	
		hints	{ }	
		Name	Type	
		base	int	
vertical_front_porch	{ }	base	int	
		hints	{ }	
		Name	Type	
		base	int	
vertical_back_porch	{ }	base	int	
		hints	{ }	
		Name	Type	
		base	int	
horizontal_sync_width	{ }	base	int	
		hints	{ }	
		Name	Type	
		base	int	
horizontal_front_porch	{ }	base	int	
		hints	{ }	
		Name	Type	
		base	int	
horizontal_back_porch	{ }	base	int	
		hints	{ }	
		Name	Type	
		base	int	

horizontal_frequency	{ }	NameType	
		Name	Type
		base	float
		hints	{ }
vertical_frequency	{ }	NameType	
		Name	Type
		base	float
		hints	{ }
pixel_rate	{ }	NameType	
		Name	Type
		base	int
		hints	{ }
scan	enum	NameType	
		Values	
		"Progressive"	
		"Interlaced"	
bits_per_component	{ }	NameType	
		Name	Type
		base	int
		hints	{ }
color_space	enum	NameType	
		Values	
		"RGB"	
		"REC709"	
signal_range	enum	NameType	
		Values	
		"0-255"	
		"16-235"	
chroma_sampling	enum	NameType	
		Values	
		"4:4:4"	
		"4:2:2"	
gamma_type	enum	NameType	
		Values	

color_primaries	enum	"POWER" "sRGB" "REC_BT1886" "SIM1" "SMPTE_ST2084" "DICOM_10" "DICOM_60" "DICOM_180" "DICOM_250" "DICOM_300" "DICOM_400"	
		Values "REC709" "REC2020" "DCI-P3-D65" "DCI-P3-Theater"	
mastering_luminance	{ }	NameType basefloat hints{ }	
		NameType	
max_c11	{ }	NameType basefloat hints{ }	
		NameType	
max_fall	{ }	NameType basefloat hints{ }	
		NameType	
content_aspect_ratio	enum	NameType	
		Values "5:4" "4:3" "16:10" "16:9" "1.85:1" "1.88:1" "1.9:1" "2.20:1"	

		"2.00:1"
		"2.35:1"
		"2.37:1"
		"2.39:1"
		"Unknown"
content_type	enum	
Values		
"Graphics"		
"Photo"		
"Cinema"		
"Game"		
"Testpattern"		
"Unknown"		
is_stereo	bool	
stereo_mode	enum	
Values		
"None"		
"Sequential"		
"FramePacked"		
"TopBottom"		
"SideBySide"		

image.connector.hdmi.edid.selected

Selected EDID for connector

MODELS All

Access: RW

Name	Type
selected	string

image.connector.hdmi.name

Name of the source

MODELS All

Access: R

Name	Type
name	string

image.connector.hdmi.signalrange

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

MODELS All

Access: RW

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

image.contrast

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

MODELS

 All

Access: RW

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

image.display.desireddisplaymode

The desired display mode.

MODELS

 All

Access: RW

Name	Type
desireddisplaymode	enum
Values	
"Mono"	
"AutoStereo"	
"ActiveStereo"	
"NightVision"	
"IGPixelShift"	
"IGPixelShiftNV"	
"IGPixelShiftFullNV"	

image.display.displaymode

The current display mode.

MODELS

 All

Access: R

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

displaymode	enum
Values	
"Mono"	
"AutoStereo"	
"ActiveStereo"	
"NightVision"	
"IGPixelShift"	
"IGPixelShiftNV"	
"IGPixelShiftFullNV"	

image.display.frequency

The display frequency.

MODELS All

Access: R

Name	Type
frequency	{ }
NameType	
base	float
hints	{ }
NameType	

image.display.synchronouslock

The display synchronous lock state.

MODELS All

Access: R

Name	Type
synchronouslock	bool

image.dynamiccontrast.level

Amount of dynamic contrast that will be applied [0:4]

MODELS All

Access: RW

Name	Type
level	enum
Values	
"OFF"	
"LOW"	
"MEDIUM"	
"HIGH"	

"MAX"

image.gamma

Image gamma. Default is 2.2

MODELS All

Access: RW

Name	Type
gamma	float
Constraints	
Minimum	1
Maximum	3
Step size	1
Precision	0.1

image.gammatype

Override the detected signal gamma type. Set to Auto for automatic control.

MODELS All

Access: RW

Name	Type
gammatype	string

image.intensity

Intensity

MODELS All

Access: RW

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

image.orientation

Description not provided

MODELS All

Access: RW

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

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

image.orientationmode

The mode of orientation as auto detected or a manual fixed setting

MODELS All

Access: RW

Name	Type
orientationmode	enum
Values	
"MANUAL "	
"AUTO_FRONT "	
"AUTO_REAR "	

image.processing.blacklevel.basicblacklevel.bottom

Bottom edge.

MODELS All

Access: RW

Name	Type
bottom	{ }
NameType	
base	int
hints	{ }
NameType	

image.processing.blacklevel.basicblacklevel.enable

Description not provided

MODELS All

Access: RW

Name	Type
enable	bool

image.processing.blacklevel.basicblacklevel.left

Left edge.

MODELS All

Access: RW

Name	Type
left	{ }
Name	Type
base	int
hints	{ }
Name	Type

image.processing.blacklevel.basicblacklevel.level

Change the upper level of the black level adjustment

MODELS All

Access: RW

Name	Type
level	int
Constraints	
Minimum	0
Maximum	65535
Step size	1
Precision	1

image.processing.blacklevel.basicblacklevel.right

Right edge.

MODELS All

Access: RW

Name	Type
right	{ }
Name	Type
base	int
hints	{ }
Name	Type

image.processing.blacklevel.basicblacklevel.top

Top edge.

MODELS All

Access: RW

Name	Type
top	{ }
Name	Type

```
base    int
hints   { }
```

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

image.processing.blacklevel.bluegain

The gain blue for black level

MODELS All

Access: RW

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

image.processing.blacklevel.file.enable

Enable/Disable black level correction

MODELS All

Access: RW

Name	Type
enable	bool

image.processing.blacklevel.file.selected

Currently selected file

MODELS All

Access: RW

Name	Type
selected	string

image.processing.blacklevel.greengain

The gain green for black level

MODELS All

Access: RW

Name	Type
greengain	float
Constraints	
Minimum	0

Maximum	1
Step size	1
Precision	0.001

image.processing.blacklevel.linkrgb

When enabled and setting one color, the others will move in the same direction

MODELS All

Access: RW

Name	Type
linkrgb	bool

image.processing.blacklevel.redgain

The gain red for black level

MODELS All

Access: RW

Name	Type
redgain	float
Constraints	
Minimum	0
Maximum	1
Step size	1
Precision	0.001

image.processing.blend.basicblend.bottom

Bottom blend edge.

MODELS All

Access: RW

Name	Type						
bottom	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>Start</td><td>{ }</td></tr></table>		Name	Type	Start	{ }		
Name	Type						
Start	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>Width</td><td>{ }</td></tr></table>		Name	Type	Width	{ }		
Name	Type						
Width	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						

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

image.processing.blend.basicblend.enable

Description not provided

MODELS All

Access: RW

Name	Type
enable	bool

image.processing.blend.basicblend.left

Left blend edge.

MODELS All

Access: RW

Name	Type						
left	{ }						
<table><thead><tr><th>Name</th><th>Type</th></tr></thead><tbody><tr><td>Start</td><td>{ }</td></tr></tbody></table>		Name	Type	Start	{ }		
Name	Type						
Start	{ }						
<table><thead><tr><th>Name</th><th>Type</th></tr></thead><tbody><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></tbody></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><thead><tr><th>Name</th><th>Type</th></tr></thead><tbody><tr><td>Width</td><td>{ }</td></tr></tbody></table>		Name	Type	Width	{ }		
Name	Type						
Width	{ }						
<table><thead><tr><th>Name</th><th>Type</th></tr></thead><tbody><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></tbody></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><thead><tr><th>Name</th><th>Type</th></tr></thead></table>		Name	Type				
Name	Type						

image.processing.blend.basicblend.right

Right blend edge.

MODELS All

Access: RW

Name	Type						
right	{ }						
<table><thead><tr><th>Name</th><th>Type</th></tr></thead><tbody><tr><td>Start</td><td>{ }</td></tr></tbody></table>		Name	Type	Start	{ }		
Name	Type						
Start	{ }						
<table><thead><tr><th>Name</th><th>Type</th></tr></thead><tbody><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></tbody></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><thead><tr><th>Name</th><th>Type</th></tr></thead></table>		Name	Type				
Name	Type						

Width { }

Name	Type
base	int
hints	{ }

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

image.processing.blend.basicblend.top

Top blend edge.

MODELS All

Access: RW

Name	Type
top	{ }

Name	Type
Start	{ }

Name	Type
base	int
hints	{ }

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

Width { }

Name	Type
base	int
hints	{ }

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

image.processing.blend.file.enable

Enable/Disable file blend

MODELS All

Access: RW

Name	Type
enable	bool

image.processing.blend.file.maxselected

Max number of selected files

MODELS All

Access: R

Name	Type
maxselected	{ }

Name	Type
base	int

hints { }

Name Type

image.processing.blend.file.selected

Currently selected files

MODELS All

Access: RW

Name	Type
selected	[string]

image.processing.blend.scurve

S-Curve exponent strength.

MODELS All

Access: RW

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

image.processing.tiltedbrightness.enable

Enable/Disable tilted brightness

MODELS All

Access: RW

Name	Type
enable	bool

image.processing.tiltedbrightness.horizontalcorrection

horizontal brightness correction

MODELS All

Access: RW

Name	Type
horizontalcorrection	int
Constraints	
Minimum	-110
Maximum	110

Step size	1
Precision	1

image.processing.tiltedbrightness.verticalcorrection

vertical brightness correction

MODELS All

Access: RW

Name	Type
verticalcorrection	int
Constraints	
Minimum	-110
Maximum	110
Step size	1
Precision	1

image.processing.transportdelay.actual

Actual transport delay.

MODELS All

Access: R

Name	Type
actual	{ }
Name Type	
base	int
hints	{ }
Name Type	

image.processing.transportdelay.desired

Desired transport delay.

MODELS All

Access: RW

Name	Type
desired	{ }
Name Type	
base	int
hints	{ }
Name Type	

image.processing.transportdelay.minimum

Minimum transport delay.

MODELS

All

Access: R

Name	Type
minimum	{ }

Name	Type
base	int
hints	{ }

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

image.processing.warp.bow.bottomleftu

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

MODELS

All

Access: RW

Name	Type
bottomleftu	{ }

Name	Type
angle	{ }

Name	Type
base	float
hints	{ }

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

| length | { } |

Name	Type
base	float
hints	{ }

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

image.processing.warp.bow.bottomleftv

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

MODELS

All

Access: RW

Name	Type
bottomleftv	{ }

Name	Type
angle	{ }

Name	Type
base	float
hints	{ }

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

length { }

Name	Type
base	float
hints	{ }

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

image.processing.warp.bow.bottomrightu

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

MODELS All

Access: RW

Name	Type
bottomrightu	{ }

Name	Type
angle	{ }

Name	Type
base	float
hints	{ }

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

length { }

Name	Type
base	float
hints	{ }

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

image.processing.warp.bow.bottomrightv

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

MODELS All

Access: RW

Name	Type
bottomrightv	{ }

Name	Type
angle	{ }

Name	Type
base	float
hints	{ }

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

length { }

Name	Type
base	float
hints	{ }

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

image.processing.warp.bow.enable

Enable/Disable bow warp

MODELS All

Access: RW

Name	Type
enable	bool

image.processing.warp.bow.symmetric

Enable/Disable symmetric mode.

MODELS All

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	{ }													
<table><tr><th>Name</th><th>Type</th></tr><tr><td>angle</td><td>{ }</td></tr></table>		Name	Type	angle	{ }									
Name	Type													
angle	{ }													
<table><tr><td>length</td><td>{ }</td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		length	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type
length	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }						
Name	Type													
base	float													
hints	{ }													
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type											
Name	Type													
<table><tr><td>length</td><td>{ }</td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		length	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type
length	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }						
Name	Type													
base	float													
hints	{ }													
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type											
Name	Type													
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type											
Name	Type													

image.processing.warp.bow.topleftv

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

MODELS All

Access: RW

Name	Type
toleftv	{ }

Name	Type
angle	{ }

Name	Type
base	float
hints	{ }

Name	Type
length	{ }

Name	Type
base	float
hints	{ }

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

image.processing.warp.bow.toprightu

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

MODELS All

Access: RW

Name	Type
toprightu	{ }

Name	Type
angle	{ }

Name	Type
base	float
hints	{ }

Name	Type
length	{ }

Name	Type
base	float
hints	{ }

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

image.processing.warp.bow.toprightv

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

MODELS All

Access: RW

Name	Type
toprightv	{ }

Name	Type
angle	{ }

		Name	Type
		base	float
		hints	{ }
		Name	Type
length		{ }	
		Name	Type
		base	float
		hints	{ }
		Name	Type

image.processing.warp.enable

Enable/Disable all warp functions

MODELS All

Access: RW

Name	Type
enable	bool

image.processing.warp.file.enable

Enable/Disable file warp

MODELS All

Access: RW

Name	Type
enable	bool

image.processing.warp.file.selected

Currently selected file

MODELS All

Access: RW

Name	Type
selected	string

image.processing.warp.fourcorners.bottomleft

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

MODELS All

Access: RW

Name	Type
bottomleft	{ }
Name	Type

x { }							
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						
y { }							
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

image.processing.warp.fourcorners.bottomright

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

MODELS

All

Access: RW

Name	Type																														
bottomright	{ }																														
<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2"> y { } </td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		Name	Type	x	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type	y { }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type
Name	Type																														
x	{ }																														
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }																								
Name	Type																														
base	int																														
hints	{ }																														
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type																												
Name	Type																														
y { }																															
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }																								
Name	Type																														
base	int																														
hints	{ }																														
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type																												
Name	Type																														

image.processing.warp.fourcorners.enable

Enable/Disable FourCorners warp

MODELS

All

Access: RW

Name	Type
enable	bool

image.processing.warp.fourcorners.screenheight

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

MODELS

All

Access: RW

Name	Type
screenheight	{ }

Name	Type
base	float
hints	{ }

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

image.processing.warp.fourcorners.screenwidth

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

MODELS All

Access: RW

Name	Type
screenwidth	{ }

Name	Type
base	float
hints	{ }

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

image.processing.warp.fourcorners.topleft

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

MODELS All

Access: RW

Name	Type
topleft	{ }

Name	Type
x	{ }

Name	Type
base	int
hints	{ }

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

y	{ }
---	-----

Name	Type
base	int
hints	{ }

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

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	{ }																														
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table></td></tr></table>	Name	Type	x	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table>	Name	Type	y	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type
Name	Type																														
x	{ }																														
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table>	Name	Type	y	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type						
Name	Type																														
base	int																														
hints	{ }																														
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table>	Name	Type	y	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type														
Name	Type																														
y	{ }																														
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																				
Name	Type																														
base	int																														
hints	{ }																														
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																												
Name	Type																														

image.processing.warp.position.enable

Enable/Disable Position warp

MODELS

 All

Access: RW

Name	Type
enable	bool

image.processing.warp.position.horizontal

The horizontal displacement factor relative to output resolution. From -HRes/2 to + HRes/2

MODELS

 All

Access: RW

Name	Type										
horizontal	{ }										
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type
Name	Type										
base	int										
hints	{ }										
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type								
Name	Type										

image.processing.warp.position.vertical

The vertical displacement factor relative to output resolution. From -VRes/2 to +VRes/2

MODELS

 All

Access: RW

Name	Type						
vertical	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						

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

image.processing.warp.warpstatus

The warp status.

MODELS All

Access: R

Name	Type
warpstatus	enum
	Values
	"Active"
	"Inactive"
	"Warning"
	"Error"

image.processing.warp.warpstatusdescription

Warp status description

MODELS All

Access: R

Name	Type
warpstatusdescription	string

image.processing.warp.zoom.enable

Enable/Disable Zoom warp

MODELS All

Access: RW

Name	Type
enable	bool

image.processing.warp.zoom.factor

The zoom factor

MODELS All

Access: RW

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

factor	float
Constraints	
Minimum	0.5
Maximum	2
Step size	1
Precision	0.1

image.processing.warp.zoom.size

Resulting zoomed resolution

MODELS All

Access: R

Name	Type																														
size	{ }																														
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table></td></tr></table>	Name	Type	x	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table>	Name	Type	y	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type
Name	Type																														
x	{ }																														
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table>	Name	Type	y	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type						
Name	Type																														
base	int																														
hints	{ }																														
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table></td></tr></table>	Name	Type	y	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type														
Name	Type																														
y	{ }																														
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																				
Name	Type																														
base	int																														
hints	{ }																														
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																												
Name	Type																														

image.resolution.alpha.size

The current resolution size (pixels x lines).

MODELS All

Access: R

Name	Type																								
size	{ }																								
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>pixels</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>lines</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr></table></td></tr></table></td></tr></table></td></tr></table>	Name	Type	pixels	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>lines</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr></table></td></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>lines</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr></table></td></tr></table>	Name	Type	lines	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr></table>	Name	Type	base	int
Name	Type																								
pixels	{ }																								
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>lines</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr></table></td></tr></table></td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>lines</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr></table></td></tr></table>	Name	Type	lines	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr></table>	Name	Type	base	int						
Name	Type																								
base	int																								
hints	{ }																								
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>lines</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr></table></td></tr></table>	Name	Type	lines	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr></table>	Name	Type	base	int														
Name	Type																								
lines	{ }																								
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr></table>	Name	Type	base	int																				
Name	Type																								
base	int																								

hints { }

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

image.resolution.availableresolutions

List all resolutions that can currently be selected.

MODELS All

Access: R

Name	Type
availableresolutions	[string]

image.resolution.beta.size

The current resolution size (pixels x lines).

MODELS All

Access: R

Name	Type																										
size	{ }																										
<table><tr><th>Name</th><th>Type</th></tr><tr><td>pixels</td><td>{ }</td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td>lines</td><td>{ }</td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>		Name	Type	pixels	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }	lines	{ }	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type
Name	Type																										
pixels	{ }																										
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }																				
Name	Type																										
base	int																										
hints	{ }																										
lines	{ }																										
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }																				
Name	Type																										
base	int																										
hints	{ }																										
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type																								
Name	Type																										

image.resolution.display.size

The current resolution size (pixels x lines).

MODELS All

Access: R

Name	Type								
size	{ }								
<table><tr><th>Name</th><th>Type</th></tr><tr><td>pixels</td><td>{ }</td></tr></table>		Name	Type	pixels	{ }				
Name	Type								
pixels	{ }								
<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }		
Name	Type								
base	int								
hints	{ }								

```
lines    { }
```

Name	Type
base	int
hints	{ }

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

image.resolution.osd.size

The current resolution size (pixels x lines).

MODELS All

Access: R

Name	Type
size	{ }
pixels	{ }
lines	{ }

Name	Type
base	int
hints	{ }

Name	Type
base	int
hints	{ }

image.resolution.output.size

The current resolution size (pixels x lines).

MODELS All

Access: R

```

Name  Type
-----
size  { }

      Name  Type
      -----
      pixels { }

                                Name  Type
                                -----
                                base   int
                                hints  { }

                                Name  Type
                                -----
                                lines  { }

                                Name  Type
                                -----
                                base   int

```

hints { }

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

image.resolution.processing.size

The current resolution size (pixels x lines).

MODELS All

Access: R

Name	Type
size	{ }

Name	Type
pixels	{ }

Name	Type
base	int
hints	{ }

Name	Type
lines	{ }

Name	Type
base	int
hints	{ }

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

image.resolution.resolution

The current resolution description.

MODELS All

Access: RW

Name	Type
resolution	string

image.resolution.size

The current resolution size (pixels x lines).

MODELS All

Access: R

Name	Type
size	{ }

Name	Type
pixels	{ }

Name	Type
base	int
hints	{ }

		Name	Type
lines		{ }	
		Name	Type
		base	int
		hints	{ }
		Name	Type

image.saturation

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

MODELS

 All

Access: RW

Name	Type
saturation	float
Constraints	
Minimum	0
Maximum	2
Step size	1
Precision	0.01

image.sharpness

Image sharpness. The value is normalized.

MODELS

 All

Access: RW

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

image.source.displayport.capabilities

Capabilities

MODELS

 All

Access: R

Name	Type
capabilities	[enum]
Values	
"HDCP"	

"HDCP2"

image.source.displayport.layout

Source layout

MODELS All

Access: R

Name	Type																																																						
layout	{ }																																																						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>rows</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>columns</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>views</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>planes</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr></table></td></tr></table></td></tr></table></td></tr></table>	Name	Type	rows	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>columns</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>views</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>planes</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr></table></td></tr></table></td></tr></table>	Name	Type	columns	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>views</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>planes</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr></table></td></tr></table>	Name	Type	views	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>planes</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr></table>	Name	Type	planes	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type																																																						
rows	{ }																																																						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																																																
Name	Type																																																						
base	int																																																						
hints	{ }																																																						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>columns</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>views</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>planes</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr></table></td></tr></table></td></tr></table>	Name	Type	columns	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>views</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>planes</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr></table></td></tr></table>	Name	Type	views	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>planes</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr></table>	Name	Type	planes	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }														
Name	Type																																																						
columns	{ }																																																						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																																																
Name	Type																																																						
base	int																																																						
hints	{ }																																																						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>views</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>planes</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr></table></td></tr></table>	Name	Type	views	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>planes</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr></table>	Name	Type	planes	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																												
Name	Type																																																						
views	{ }																																																						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																																																
Name	Type																																																						
base	int																																																						
hints	{ }																																																						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>planes</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr></table>	Name	Type	planes	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																																										
Name	Type																																																						
planes	{ }																																																						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																																																
Name	Type																																																						
base	int																																																						
hints	{ }																																																						

image.source.displayport.name

Name of the source

MODELS All

Access: R

Name	Type
name	string

image.source.hdmi.capabilities

Capabilities

MODELS All

Access: R

Name	Type
capabilities	[enum]
Values	
"HDCP"	
"HDCP2"	

image.source.hdmi.layout

Source layout

MODELS All

Access: R

Name	Type
layout	{ }
NameType	
rows	{ }
NameType	
base	int
hints	{ }
NameType	
columns	{ }
NameType	
base	int
hints	{ }
NameType	
views	{ }
NameType	
base	int
hints	{ }
NameType	
planes	{ }
NameType	
base	int
hints	{ }
NameType	

image.source.hdmi.name

Name of the source

MODELS All

Access: R

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

name

string

image.stereo.darktime

Darktime in us.

MODELS

All

Access: RW

Name	Type						
darktime	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

image.stereo.glassync.delay

Sync delay in us.

MODELS

All

Access: RW

Name	Type						
delay	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

image.stereo.glassync.delaymaximum

Maximum sync delay in us.

MODELS

All

Access: R

Name	Type						
delaymaximum	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

image.stereo.glassync.delayminimum

Minimum sync delay in us.

MODELS

All

Access: R

Name	Type
delayminimum	{ }

Name	Type
base	int
hints	{ }

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

image.stereo.glassync.invert

Sync invert.

MODELS All

Access: RW

Name	Type
invert	bool

image.stereo.swapframepair

swap which stereo frames belong to each other

MODELS All

Access: RW

Name	Type
swapframepair	bool

image.testpattern.available

When false, the service will deny outside property change requests

MODELS All

Access: R

Name	Type
available	bool

image.testpattern.enablefilemanagement

Enables or disables the ability to upload download and delete custom testpatterns

MODELS All

Access: R

Name	Type
enablefilemanagement	bool

image.testpattern.iterator.calibration.list

The list of testpatterns to iterate in the viewing order.

MODELS

All

Access: R

Name	Type
list	[string]

image.testpattern.iterator.calibration.size

Return the size of the list of test patterns to iterate upon.

MODELS

All

Access: R

Name	Type						
size	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

image.testpattern.iterator.iris.list

The list of testpatterns to iterate in the viewing order.

MODELS

All

Access: R

Name	Type
list	[string]

image.testpattern.iterator.iris.size

Return the size of the list of test patterns to iterate upon.

MODELS

All

Access: R

Name	Type						
size	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

image.testpattern.iterator.optics.list

The list of testpatterns to iterate in the viewing order.

MODELS

All

Access: R

Name	Type
list	[string]

image.testpattern.iterator.optics.size

Return the size of the list of test patterns to iterate upon.

MODELS All

Access: R

Name	Type						
size	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						

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

image.testpattern.iterator.shift.list

The list of testpatterns to iterate in the viewing order.

MODELS All

Access: R

Name	Type
list	[string]

image.testpattern.iterator.shift.size

Return the size of the list of test patterns to iterate upon.

MODELS All

Access: R

Name	Type						
size	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						

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

image.testpattern.iterator.stereo.list

The list of testpatterns to iterate in the viewing order.

MODELS All

Access: R

Name	Type
list	[string]

image.testpattern.iterator.stereo.size

Return the size of the list of test patterns to iterate upon.

MODELS All

Access: R

Name	Type
size	{ }

Name	Type
base	int
hints	{ }

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

image.testpattern.iterator.xpr.list

The list of testpatterns to iterate in the viewing order.

MODELS All

Access: R

Name	Type
list	[string]

image.testpattern.iterator.xpr.size

Return the size of the list of test patterns to iterate upon.

MODELS All

Access: R

Name	Type
size	{ }

Name	Type
base	int
hints	{ }

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

image.testpattern.nosignal.pattern

The unique test pattern id to show as the NoSignal pattern

MODELS All

Access: RW

Name	Type
pattern	string

image.testpattern.selected

The unique id of the selected pattern

MODELS

 All

Access: RW

Name	Type
selected	string

image.testpattern.show

Enable/disable the display of the selected pattern

MODELS

 All

Access: RW

Name	Type
show	bool

image.window.main.activesignal

Indicates whether or not the currently selected source has an active signal (true) or not (false)

MODELS

 All

Access: R

Name	Type
activesignal	bool

image.window.main.position

Window position

MODELS

 All

Access: R

Name	Type																										
position	{ }																										
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td>y</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>	Name	Type	x	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	y	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type
Name	Type																										
x	{ }																										
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																				
Name	Type																										
base	int																										
hints	{ }																										
y	{ }																										
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																				
Name	Type																										
base	int																										
hints	{ }																										
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																								
Name	Type																										

image.window.main.scalingmode

The scaling mode to apply to the source

MODELS All

Access: RW

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

image.window.main.size

Window size

MODELS All

Access: R

Name	Type
size	{ }
NameType	
width	{ }
NameType	
base	int
hints	{ }
NameType	
height	{ }
NameType	
base	int
hints	{ }
NameType	

image.window.main.source

The source that is displayed in this window

MODELS All

Access: RW

Name	Type
source	string

image.window.main.sourceinvalidityreasons

The reason(s) the source is not valid

MODELS

 All

Access: R

Name	Type
sourceinvalidityreasons	[enum]
	Values
	"ResolutionOutsideFlexRange"

iot.certificate.applicable

True if the certificate applies to this projector

MODELS

 All

Access: R

Name	Type
applicable	bool

iot.certificate.available

True if a certificate was found

MODELS

 All

Access: R

Name	Type
available	bool

iot.certificate.issuer

certificate Issuer info

MODELS

 All

Access: R

Name	Type
issuer	string

iot.certificate.valid

Result of the validity check of the certificate

MODELS

 All

Access: R

Name	Type
valid	bool

iot.certificate.validityend

certificate Valid until this date

MODELS

 All

Access: R

Name	Type
validityend	string

iot.certificate.validitystart

certificate Valid from this date

MODELS

 All

Access: R

Name	Type
validitystart	string

iot.cloudservices.connectionstate

The current state of the connection.

MODELS

 All

Access: R

Name	Type
connectionstate	enum
Values	
"CONNECTED"	
"DISCONNECTED"	
"CONNECTING"	
"DISCONNECTING"	
"FAILED"	

iot.cloudservices.enable

True if IOT cloud connection is enabled, false otherwise

MODELS

 All

Access: RW

Name	Type
enable	bool

iot.proxy.address

This is the address (hostname/ip) of the proxy.

MODELS

 All

Access: RW

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

addressstring

iot.proxy.enabled

When enabled, iot connection will go over proxy specified in othe proxy parameters

MODELSAll

Access: RW

Name	Type
enabled	bool

iot.proxy.password

Password used to log in to proxy.

MODELSAll

Access: RW

Name	Type
password	string

iot.proxy.port

The port to use for the proxy connection.

MODELSAll

Access: RW

Name	Type
port	{ }
Name	Type
base	int
hints	{ }
Name	Type

iot.proxy.username

Username used to log in to proxy.

MODELSAll

Access: RW

Name	Type
username	string

iot.requiretermagreement

Indiates whether or not the user is required to accept the term agreement.

MODELSAll

Access: R

Name	Type
requiretermagreement	bool

iot.termagreement

Represents the users acknowledge of the IOT terms of agreement.

MODELS All

Access: RW

Name	Type
termagreement	bool

keydispatcher.repeatdelay

Initial key repeat delay in milliseconds

MODELS All

Access: R

Name	Type
repeatdelay	{ }
Name	Type
base	int
hints	{ }
Name	Type

keydispatcher.repeatinterval

Key repeat interval in milliseconds

MODELS All

Access: R

Name	Type
repeatinterval	{ }
Name	Type
base	int
hints	{ }
Name	Type

network.device.lan.carrier

Whether the device has carrier or not

MODELS All

Access: R

Name	Type
carrier	bool

network.device.lan.configuration

The configuration method of the device: auto or manual

MODELS

All

Access: RW

Name	Type
configuration	enum
Values	
"AUTO"	
"MANUAL"	

network.device.lan.devicetype

The general type of the network device

MODELS

All

Access: R

Name	Type
devicetype	enum
Values	
"WIRED"	
"WIRELESS"	
"MOBILE"	

network.device.lan.hwaddress

The active hardware (MAC) address

MODELS

All

Access: R

Name	Type
hwaddress	string

network.device.lan.ip4config

The current configuration for IP version 4

MODELS

All

Access: R

Name	Type
ip4config	{ }
Name	Type

Address	string
Mask	string
Gateway	string
NameServers	string

network.device.lan.ip4configmanual

Get/set the manual configuration for IP version 4

MODELS All

Access: RW

Name	Type
ip4configmanual	{ }

Name	Type
Address	string
Mask	string
Gateway	string
NameServers	string

network.device.lan.ip6config

The current configuration for IP version 6

MODELS All

Access: R

Name	Type
ip6config	{ }

Name	Type
Address	string
Prefix	string
Gateway	string
NameServers	string

network.device.lan.ip6configmanual

Get/set the manual configuration for IP version 4

MODELS All

Access: RW

Name	Type
ip6configmanual	{ }

Name	Type
Address	string
Prefix	string
Gateway	string

NameServers string

network.device.lan.speed

The speed of the device in Mbit/s

MODELS All

Access: R

Name	Type
speed	{ }
Name	Type
base	int
hints	{ }
Name	Type

network.device.lan.state

The current state of the device

MODELS All

Access: R

Name	Type
state	enum
Values	
"CONNECTED"	
"DISCONNECTED"	
"CONNECTING"	
"NEEDAUTH"	
"IPCONFIG"	
"IPCHECK"	
"DISCONNECTING"	
"FAILED"	

network.device.lan.stateinfo

Additional information about the device state. Can be empty

MODELS All

Access: R

Name	Type
stateinfo	string

network.domain

The domain name

MODELS All

Access: RW

Name	Type
domain	string

network.hostname

The host name

MODELS All

Access: RW

Name	Type
hostname	string

network.version

The Networking Service version

MODELS All

Access: R

Name	Type
version	string

notification.count

The number of notifications received and dismissed

MODELS All

Access: R

Name	Type						
count	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

optics.focus.calibrationstate

Current calibration state

MODELS All

Access: R

Name	Type			
calibrationstate	enum			
	<table><tr><th>Values</th></tr><tr><td>"Unknown"</td></tr><tr><td>"Ok"</td></tr></table>	Values	"Unknown"	"Ok"
Values				
"Unknown"				
"Ok"				

"Busy"

"Error"

"NotImplemented"

optics.focus.enabled

Enabled state

MODELS All

Access: RW

Name	Type
enabled	bool

optics.focus.limits.forward

Forward limit reached

MODELS All

Access: R

Name	Type
forward	bool

optics.focus.limits.reverse

Reverse limit reached

MODELS All

Access: R

Name	Type
reverse	bool

optics.focus.maxposition

Maximum available position

MODELS All

Access: R

Name	Type						
maxposition	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

optics.focus.minposition

Minimum available position

MODELS

 All

Access: R

Name	Type
minposition	{ }
Name	Type
base	int
hints	{ }
Name	Type

optics.focus.position

Current position

MODELS

 All

Access: R

Name	Type
position	{ }
Name	Type
base	int
hints	{ }
Name	Type

optics.focus.safetocalibrate

Safe to calibrate

MODELS

 All

Access: R

Name	Type
safetocalibrate	bool

optics.focus.safetooperate

Safe to operate state

MODELS

 All

Access: R

Name	Type
safetooperate	bool

optics.focus.state

Current state

MODELS

 All

Access: R

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

optics.focus.target

Desired target

MODELS All

Access: RW

Name	Type
target	{ }
NameType	
base	int
hints	{ }
NameType	

optics.irisoperation

Property for selecting whether iris operation should be linked or individual

MODELS All

Access: RW

Name	Type
irisoperation	enum
Values	
"Individual"	
"Linked"	

optics.irisoperationavailable

Description not provided

MODELS All

Access: R

Name	Type
irisoperationavailable	bool

optics.lens.lens

Currently selected lens

MODELS All

Access: R

Name	Type
lens	{ }
Name	Type
Id	string
Description	string

optics.lens.metadata

Current lens metadata

MODELS All

Access: R

Name	Type
metadata	[{ }]
	{ }
Name	Type
key	string
value	string

optics.lens.position.list

List of available lens positions

MODELS All

Access: R

Name	Type
list	[string]

optics.lens.position.selected

Currently selected lens position

MODELS All

Access: RW

Name	Type
selected	string

optics.lens.position.test1.focus

Lens positions stored focus position

MODELS All

Access: R

Name	Type
focus	{ }

Name	Type
base	int
hints	{ }

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

optics.lens.position.test1.name

Lens positions name

MODELS All

Access: R

Name	Type
name	string

optics.lens.position.test1.shifthorizontal

Lens positions stored horizontal shift position

MODELS All

Access: R

Name	Type
shifthorizontal	{ }

Name	Type
base	int
hints	{ }

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

optics.lens.position.test1.shiftvertical

Lens positions stored vertical shift position

MODELS All

Access: R

Name	Type
shiftvertical	{ }

Name	Type
base	int
hints	{ }

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

optics.lens.position.test1.status

Lens position status

MODELS All

Access: R

Name	Type
status	enum
Values	
"Inactive"	
"Activating"	
"Active"	
"Error"	

optics.lens.position.test1.zoom

Lens positions stored zoom position

MODELS All

Access: R

Name	Type
zoom	{ }
Name Type	
base	int
hints	{ }
Name Type	

optics.lens.throwratio

Throw ratios of the selected lens

MODELS All

Access: R

Name	Type
throwratio	[{ }]
{ }	
Name Type	
key	string
value	{ }
Name Type	
Min	{ }
Name Type	
base	float
hints	{ }
Name Type	
Max	{ }
Name Type	
base	float
hints	{ }
Name Type	

optics.lenspresent

Lens present

MODELS All

Access: R

Name	Type
lenspresent	bool

optics.lensshift.horizontal.calibrationstate

Current calibration state

MODELS All

Access: R

Name	Type
calibrationstate	enum
Values	
"Unknown"	
"Ok"	
"Busy"	
"Error"	
"NotImplemented"	

optics.lensshift.horizontal.enabled

Enabled state

MODELS All

Access: RW

Name	Type
enabled	bool

optics.lensshift.horizontal.limits.forward

Forward limit reached

MODELS All

Access: R

Name	Type
forward	bool

optics.lensshift.horizontal.limits.reverse

Reverse limit reached

MODELS All

Access: R

Name	Type
reverse	bool

optics.lensshift.horizontal.maxposition

Maximum available position

MODELS All

Access: R

Name	Type
maxposition	{ }

Name	Type
base	int
hints	{ }

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

optics.lensshift.horizontal.minposition

Minimum available position

MODELS All

Access: R

Name	Type
minposition	{ }

Name	Type
base	int
hints	{ }

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

optics.lensshift.horizontal.position

Current position

MODELS All

Access: R

Name	Type
position	{ }

Name	Type
base	int
hints	{ }

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

optics.lensshift.horizontal.safetocalibrate

Safe to calibrate

MODELS

All

Access: R

Name	Type
safetocalibrate	bool

optics.lensshift.horizontal.safetooperate

Safe to operate state

MODELS

All

Access: R

Name	Type
safetooperate	bool

optics.lensshift.horizontal.state

Current state

MODELS

All

Access: R

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

optics.lensshift.horizontal.target

Desired target

MODELS

All

Access: RW

Name	Type
target	{ }
NameType	
base	int
hints	{ }
NameType	

optics.lensshift.vertical.calibrationstate

Current calibration state

MODELS

All

Access: R

Name	Type
calibrationstate	enum
Values	
"Unknown"	
"Ok"	
"Busy"	
"Error"	
"NotImplemented"	

optics.lensshift.vertical.enabled

Enabled state

MODELS All

Access: RW

Name	Type
enabled	bool

optics.lensshift.vertical.limits.forward

Forward limit reached

MODELS All

Access: R

Name	Type
forward	bool

optics.lensshift.vertical.limits.reverse

Reverse limit reached

MODELS All

Access: R

Name	Type
reverse	bool

optics.lensshift.vertical.maxposition

Maximum available position

MODELS All

Access: R

Name	Type
maxposition	{ }
Name Type	

base	int
hints	{ }

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

optics.lensshift.vertical.minposition

Minimum available position

MODELS All

Access: R

Name	Type
minposition	{ }

base	int
hints	{ }

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

optics.lensshift.vertical.position

Current position

MODELS All

Access: R

Name	Type
position	{ }

base	int
hints	{ }

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

optics.lensshift.vertical.safetocalibrate

Safe to calibrate

MODELS All

Access: R

Name	Type
safetocalibrate	bool

optics.lensshift.vertical.safetooperate

Safe to operate state

MODELS All

Access: R

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

safetooperate

bool

optics.lensshift.vertical.state

Current state

MODELS

All

Access: R

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

optics.lensshift.vertical.target

Desired target

MODELS

All

Access: RW

Name	Type
target	{ }
NameType	
base	int
hints	{ }
NameType	

optics.shutter.position

Position of shutter

MODELS

All

Access: R

Name	Type
position	enum
Values	
"Open"	
"Closed"	

optics.shutter.target

Target position of shutter

MODELS

All

Access: RW

Name	Type
target	enum
Values	
"Open"	
"Closed"	

optics.zoom.calibrationstate

Current calibration state

MODELS All

Access: R

Name	Type
calibrationstate	enum
Values	
"Unknown"	
"Ok"	
"Busy"	
"Error"	
"NotImplemented"	

optics.zoom.enabled

Enabled state

MODELS All

Access: RW

Name	Type
enabled	bool

optics.zoom.limits.forward

Forward limit reached

MODELS All

Access: R

Name	Type
forward	bool

optics.zoom.limits.reverse

Reverse limit reached

MODELS All

Access: R

Name	Type
reverse	bool

optics.zoom.maxposition

Maximum available position

MODELS All

Access: R

Name	Type						
maxposition	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

optics.zoom.minposition

Minimum available position

MODELS All

Access: R

Name	Type						
minposition	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

optics.zoom.position

Current position

MODELS All

Access: R

Name	Type						
position	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

optics.zoom.safetocalibrate

Safe to calibrate

MODELS All

Access: R

Name	Type
safetocalibrate	bool

optics.zoom.safetooperate

Safe to operate state

MODELS All

Access: R

Name	Type
safetooperate	bool

optics.zoom.state

Current state

MODELS All

Access: R

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

optics.zoom.target

Desired target

MODELS All

Access: RW

Name	Type
target	{ }
Name Type	
base	int
hints	{ }
Name Type	

profile.domains

The list of available domains for profiles.

MODELS All

Access: R

Name	Type
domains	[string]

profile.presetassignments

All preset assignments.

MODELS All

Access: R

Name	Type														
presetassignments	[{ }]														
	{ }														
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>key</td><td>{ }</td></tr></table>	Name	Type	key	{ }										
Name	Type														
key	{ }														
	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>value</td><td>string</td></tr></table></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>value</td><td>string</td></tr></table>	Name	Type	value	string
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }								
Name	Type														
base	int														
hints	{ }														
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>value</td><td>string</td></tr></table>	Name	Type	value	string										
Name	Type														
value	string														

profile.profiles

The list of created profiles.

MODELS All

Access: R

Name	Type
profiles	[string]

property.notifyself

Description not provided

MODELS All

Access: RW

Name	Type
notifyself	bool

protocols.pjlink.authenticationrequired

Enable/disable PJLink authentication procedure

MODELS All

Access: R

Name	Type
authenticationrequired	bool

protocols.pjlink.enable

Enable/disable PJLink

MODELS All

Access: R

Name	Type
enable	bool

remotecontrol.address

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

MODELS All

Access: RW

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

remotecontrol.broadcastaddress

The broadcast address

MODELS All

Access: RW

Name	Type
broadcastaddress	int
Constraints	
Minimum	0
Maximum	1
Step size	1
Precision	1

remotecontrol.sensors.front.enable

Enable or disable the IR sensor

MODELS All

Access: RW

Name	Type
enable	bool

remotecontrol.sensors.front.name

The display name of the IR sensor

MODELS All

Access: R

Name	Type
name	string

remotecontrol.sensors.rear.enable

Enable or disable the IR sensor

MODELS All

Access: RW

Name	Type
enable	bool

remotecontrol.sensors.rear.name

The display name of the IR sensor

MODELS All

Access: R

Name	Type
name	string

remotecontrol.sensors.side.enable

Enable or disable the IR sensor

MODELS All

Access: RW

Name	Type
enable	bool

remotecontrol.sensors.side.name

The display name of the IR sensor

MODELS All

Access: R

Name	Type
name	string

remotecontrol.sensors.wiredremote-layer1.enable

Enable or disable the IR sensor

MODELS

All

Access: RW

Name	Type
enable	bool

remotecontrol.sensors.wiredremote-layer1.name

The display name of the IR sensor

MODELS

All

Access: R

Name	Type
name	string

remotecontrol.sensors.wiredremote.enable

Enable or disable the IR sensor

MODELS

All

Access: RW

Name	Type
enable	bool

remotecontrol.sensors.wiredremote.name

The display name of the IR sensor

MODELS

All

Access: R

Name	Type
name	string

scheduler.actions

List of all scheduled actions' name.

MODELS

All

Access: R

Name	Type
actions	[string]

scheduler.actiontypes

List of all available action types.

MODELS

All

Access: R

Name	Type
actiontypes	[string]

scheduler.nextscheduledaction

The next task(s) that will be executed. (Can be more than one task if multiple tasks have the exact same execution time.)

MODELS All

Access: R

Name	Type
nextscheduledaction	[{ }]

{ }

Name	Type
Name	string
Type	string
Transition	string
Macro	string
Profile	string
LaunchTimePoint	string

scheduler.pause

Return True if action is paused, otherwise return False.

MODELS All

Access: RW

Name	Type
pause	bool

scheduler.scheduledactions

List of all scheduled actions data.

MODELS All

Access: RW

Name	Type
scheduledactions	string

scheduler.transitions

List of all available transitions for system action.

MODELS All

Access: R

Name	Type
transitions	[string]

screen.hdrboost

[DEPRECATED] Replaced by image.processing.colormapping.hdrboost

MODELS

 All

Access: RW

Name	Type
hdrboost	float
Constraints	
Minimum	0.5
Maximum	2
Step size	0.01
Precision	0.1

screen.luminance

The maximum luminance measured on the screen in cd/m2

MODELS

 All

Access: RW

Name	Type
luminance	float
Constraints	
Minimum	50
Maximum	10000
Step size	10
Precision	1

screen.screenaspect

The calculated aspect of the ScreenWidth/ScreenHeight

MODELS

 All

Access: R

Name	Type
screenaspect	{ }
Name	Type
base	float
hints	{ }
Name	Type

screen.screensize

The width and height of the screen we are projecting on.

MODELS

 All

Access: RW

Name	Type
screensize	{ }

Name	Type
width	{ }

Name	Type
base	float
hints	{ }

Name	Type
height	{ }

Name	Type
base	float
hints	{ }

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

statistics.illumination on uptime.value

Counter value

MODELS

 All

Access: R

Name	Type
value	{ }

Name	Type
base	int
hints	{ }

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

statistics.laser runtime.value

Counter value

MODELS

 All

Access: R

Name	Type
value	{ }

Name	Type
base	int
hints	{ }

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

statistics.laser.plate01.bank01.runtimeseconds.value

Counter value

MODELS All

Access: R

Name	Type
value	{ }
Name	Type
base	int
hints	{ }
Name	Type

statistics.laser.plate01.bank02.runtimeseconds.value

Counter value

MODELS All

Access: R

Name	Type
value	{ }
Name	Type
base	int
hints	{ }
Name	Type

statistics.laser.plate02.bank01.runtimeseconds.value

Counter value

MODELS All

Access: R

Name	Type
value	{ }
Name	Type
base	int
hints	{ }
Name	Type

statistics.laser.plate02.bank02.runtimeseconds.value

Counter value

MODELS All

Access: R

Name	Type
value	{ }
Name	Type
base	int
hints	{ }

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

statistics.maintenance.actuator.value

Counter value

MODELS All

Access: R

Name	Type
value	{ }
Name	Type
base	int
hints	{ }

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

statistics.maintenance.colorwheel.value

Counter value

MODELS All

Access: R

Name	Type
value	{ }
Name	Type
base	int
hints	{ }

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

statistics.maintenance.phosphorwheel.value

Counter value

MODELS All

Access: R

Name	Type
value	{ }
Name	Type
base	int
hints	{ }

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

statistics.projector on.value

Counter value

MODELS All

Access: R

Name	Type
value	{ }

Name	Type
base	int
hints	{ }

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

statistics.projector uptime.value

Counter value

MODELS All

Access: R

Name	Type
value	{ }

Name	Type
base	int
hints	{ }

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

statistics.projector.value

Counter value

MODELS All

Access: R

Name	Type
value	{ }

Name	Type
base	int
hints	{ }

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

system.articlenumber

Article number.

MODELS All

Access: R

Name	Type
articlenumber	string

system.colorwheel

Article number of installed color wheel

MODELS All

Access: R

Name	Type
colorwheel	string

system.colorwheelname

Name of installed color wheel

MODELS All

Access: R

Name	Type
colorwheelname	string

system.date.availabletimezones

An array of all available time zones and their information.

MODELS All

Access: R

Name	Type				
availabletimezones	[{ }]				
	{ }				
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>name</td><td>string</td></tr></table>	Name	Type	name	string
Name	Type				
name	string				

system.date.ntp.connected

True if there is a connection to the NTP-server.

MODELS All

Access: R

Name	Type
connected	bool

system.date.ntp.enabled

True if NTP time synchronization should be used.

MODELS All

Access: R

Name	Type
enabled	bool

system.date.ntp.server

The NTP server hostname or address.

MODELS All

Access: R

Name	Type
server	string

system.date.timezone

The configured time zone of the system.

MODELS All

Access: R

Name	Type
timezone	string

system.eco.available

Returns true if state is available for this projector

MODELS All

Access: R

Name	Type
available	bool

system.eco.enable

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

MODELS All

Access: RW

Name	Type
enable	bool

system.familyname

Family name.

MODELS All

Access: R

Name	Type
familyname	string

system.firmwareversion

Firmware version.

MODELS All

Access: R

Name	Type
firmwareversion	string

system.health

The current global health state of the projector. State error means the projector can not safely be operated. Warning means the show can go on, but it is strongly advised to find the cause of the warning and solve it. Normal means the projector is performing normally.

MODELS All

Access: R

Name	Type
health	enum
Values	
"Normal"	
"Warning"	
"Error"	

system.initialstate

State to transition to when the unit is started

MODELS All

Access: RW

Name	Type
initialstate	enum
Values	
"boot"	
"eco"	
"standby"	
"ready"	
"conditioning"	
"on"	
"service"	
"deconditioning"	
"error"	

system.license.applicable

Applicability of the license file.

MODELS All

Access: R

Name	Type
applicable	bool

system.license.available

Availability of a license file.

MODELS All

Access: R

Name	Type
available	bool

system.license.options

A dictionary of options and their values.

MODELS All

Access: R

Name	Type
options	[{ }]

{ }

Name	Type
key	string
value	string

system.license.register.mandatory

Shows if registering the product is mandatory or optional

MODELS All

Access: R

Name	Type
mandatory	bool

system.license.register.valid

True when the product has been successfully registered.

MODELS All

Access: R

Name	Type
valid	bool

system.license.valid

Validity of the license file.

MODELS All

Access: R

Name	Type
valid	bool

system.modelname

Model name.

MODELS All

Access: R

Name	Type
modelName	string

system.name

Custom name for this device.

MODELS All

Access: R

Name	Type
name	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
Constraints	
Minimum	60
Maximum	7200
Step size	30
Precision	1

system.on.timeout.enable

Enable/disable the timeout.

MODELS

All

Access: RW

Name	Type
enable	bool

system.on.timeout.remaining

The remaining amount of seconds before the timer will timeout.

MODELS

All

Access: R

Name	Type
remaining	{ }

Name	Type
base	int
hints	{ }

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

system.operationalmodes.availablemodes

List of available operational modes

MODELS

All

Access: R

Name	Type
availablemodes	[enum]

Values
"normal"
"silent"
"longlife"
"highambienttemperature"

system.operationalmodes.mode

The currently active operational mode

MODELS

All

Access: RW

Name	Type
mode	enum

Values
"normal"
"silent"
"longlife"

"highambienttemperature"

system.ready.timeout.duration

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

MODELS All

Access: RW

Name	Type
duration	int
Constraints	
Minimum	60
Maximum	7200
Step size	30
Precision	1

system.ready.timeout.enable

Enable/disable the timeout.

MODELS All

Access: RW

Name	Type
enable	bool

system.ready.timeout.remaining

The remaining amount of seconds before the timer will timeout.

MODELS All

Access: R

Name	Type
remaining	{ }
NameType	
base	int
hints	{ }
NameType	

system.resetprogress

Reset progress: [0..100]

MODELS All

Access: R

Name	Type
resetprogress	{ }

Name	Type
base	int
hints	{ }

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

system.resetresult

Description not provided

MODELS All

Access: R

Name	Type
resetresult	{ }

Name	Type
time	string
result	enum

Values
"UNKNOWN"
"SUCCESS"
"FAILED"

system.resetstatus

Reset status

MODELS All

Access: R

Name	Type
resetstatus	enum

Values
"Idle"
"Resetting"
"Failed"
"Finished"

system.serialnumber

Serial number.

MODELS All

Access: R

Name	Type
serialnumber	string

system.standby.available

Returns true if state is available for this projector

MODELS

 All

Access: R

Name	Type
available	bool

system.standby.enable

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

MODELS

 All

Access: RW

Name	Type
enable	bool

system.standby.timeout.duration

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

MODELS

 All

Access: RW

Name	Type
duration	int
Constraints	
Minimum	60
Maximum	7200
Step size	30
Precision	1

system.standby.timeout.enable

Enable/disable the timeout.

MODELS

 All

Access: RW

Name	Type
enable	bool

system.standby.timeout.remaining

The remaining amount of seconds before the timer will timeout.

MODELS

 All

Access: R

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

remaining { }							
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

system.state

The current state of the unit

MODELS All

Access: R

Name	Type
state	enum
Values	
"boot"	
"eco"	
"standby"	
"ready"	
"conditioning"	
"on"	
"service"	
"deconditioning"	
"error"	

system.targetstate

The state the unit is transitioning into

MODELS All

Access: R

Name	Type
targetstate	enum
Values	
"boot"	
"eco"	
"standby"	
"ready"	
"conditioning"	
"on"	
"service"	
"deconditioning"	
"error"	

ui.access.enduser

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

MODELS

 All

Access: R

Name	Type
enduser	bool

ui.backlight.state

Description not provided

MODELS

 All

Access: RW

Name	Type
state	enum
Values	
"off"	
"on"	
"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
Constraints	
Minimum	5
Maximum	3600
Step size	1
Precision	1

ui.hasstealthmode

Description not provided

MODELS

 All

Access: R

Name	Type
hasstealthmode	bool

ui.language

The user interface language

MODELS

All

Access: RW

Name	Type
language	string

ui.layer.advancedblend.drawing

Drawing commands in the form of a JSON object

MODELS

All

Access: RW

Name	Type
drawing	string

ui.layer.advancedblend.enable

Enable or disable the layer

MODELS

All

Access: RW

Name	Type
enable	bool

ui.layer.advancedblend.palette

Color palette that can be used when drawing the blend layer

MODELS

All

Access: RW

Name	Type
palette	[string]

ui.layer.basicblacklevel.color

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

MODELS

All

Access: RW

Name	Type
color	string

ui.layer.basicblacklevel.enable

Enable or disable the layer

MODELS

All

Access: RW

Name	Type
enable	bool

ui.layer.basicblacklevel.selection

Toggle edge selection

MODELS All

Access: RW

Name	Type
selection	{ }

Name	Type
Bottom	bool
Left	bool
Right	bool
Top	bool

ui.layer.basicblacklevel.selectioncolor

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

MODELS All

Access: RW

Name	Type
selectioncolor	string

ui.layer.basicblend.color

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

MODELS All

Access: RW

Name	Type
color	string

ui.layer.basicblend.enable

Enable or disable the layer

MODELS All

Access: RW

Name	Type
enable	bool

ui.layer.basicblend.selection

Toggle edge selection

MODELS All

Access: RW

Name	Type
selection	{ }

Name	Type
Bottom	bool
Left	bool
Right	bool
Top	bool

ui.layer.basicblend.selectioncolor

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

MODELS All

Access: RW

Name	Type
selectioncolor	string

ui.layer.blank.border

(Optional) Show/hide a border/outline of the screen

MODELS All

Access: RW

Name	Type
border	bool

ui.layer.blank.bordercolor

(Optional) Specify the border color, e.g '#ff0000' or 'rgba(255,0,0,0.75)'

MODELS All

Access: RW

Name	Type
bordercolor	string

ui.layer.blank.enable

Enable or disable the layer. When enabled, the screen will be covered in black.

MODELS All

Access: RW

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

<code>enable</code>	<code>bool</code>
---------------------	-------------------

ui.layer.blank.icon

A single UTF character from the Barco-Icons font set.

MODELS

All

Access: RW

Name	Type
<code>icon</code>	<code>string</code>

ui.layer.blank.show

Used only for persisting whether or not to show the layer

MODELS

All

Access: RW

Name	Type
<code>show</code>	<code>bool</code>

ui.layer.blank.showtext

Show or hide the text

MODELS

All

Access: RW

Name	Type
<code>showtext</code>	<code>bool</code>

ui.layer.blank.text

(Optional) Specify a text to show at the center of the screen

MODELS

All

Access: RW

Name	Type
<code>text</code>	<code>string</code>

ui.layer.fourcorner.cornercolor

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

MODELS

All

Access: RW

Name	Type
<code>cornercolor</code>	<code>string</code>

ui.layer.fourcorner.enable

Enable or disable the layer

MODELS All

Access: RW

Name	Type
enable	bool

ui.layer.fourcorner.linecolor

The line color, e.g '#ffff00' or '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

MODELS All

Access: RW

Name	Type
lines	bool

ui.layer.fourcorner.selection

Toggle corner selection

MODELS All

Access: RW

Name	Type										
selection	{ }										
<table><tr><th>Name</th><th>Type</th></tr><tr><td>TopLeft</td><td>bool</td></tr><tr><td>TopRight</td><td>bool</td></tr><tr><td>BottomRight</td><td>bool</td></tr><tr><td>BottomLeft</td><td>bool</td></tr></table>		Name	Type	TopLeft	bool	TopRight	bool	BottomRight	bool	BottomLeft	bool
Name	Type										
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)'

MODELS All

Access: RW

Name	Type
selectioncolor	string

ui.layer.grid.color

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

MODELS All

Access: RW

Name	Type
color	string

ui.layer.grid.enable

Enable or disable the layer

MODELS All

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
mark	[{ }]
	{ }

Name	Type
X	{ }

Name	Type
base	int
hints	{ }

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

Y { }

Name	Type
base	int
hints	{ }

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

Color string

ui.layer.grid.points

Number of grid points

MODELS All

Access: RW

Name	Type
points	enum
Values	
"2x2 "	
"3x3 "	
"5x5 "	
"9x9 "	
"17x17 "	
"33x33 "	

ui.layer.grid.showlines

Toggle drawing lines between grid points

MODELS All

Access: RW

Name	Type
showlines	bool

ui.layer.grid.showpoints

Toggle drawing grid points

MODELS All

Access: RW

Name	Type
showpoints	bool

ui.length

Preferred unit for display of lengths and distances

MODELS All

Access: RW

Name	Type
length	enum
Values	
"M"	
"CM"	
"FT"	
"IN"	

ui.lensmenu

Show or hide the lens menu

MODELS All

Access: RW

Name	Type
lensmenu	bool

ui.luminance

Preferred unit for luminance

MODELS All

Access: RW

Name	Type
luminance	enum
Values	
"NITS"	
"FL"	

ui.menu

Show or hide the menu

MODELS All

Access: RW

Name	Type
menu	bool

ui.menuposition

Placement of menu related to full screen.

MODELS All

Access: RW

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

`menuposition` enum

Values

- "TopLeft"
- "Top"
- "TopRight"
- "Right"
- "BottomRight"
- "Bottom"
- "BottomLeft"
- "Left"
- "Center"

ui.message

The persistent message to show on the screen. Rich text allowed

MODELS All

Access: R

Name	Type
message	{ }
title	string
icon	string
message	string

ui.messageposition

Placement of the message relative to full screen

MODELS All

Access: R

Name	Type
messageposition	enum
Values	
"TopLeft"	
"Top"	
"TopRight"	
"Right"	
"BottomRight"	
"Bottom"	
"BottomLeft"	
"Left"	
"Center"	

ui.minimize

Minimize the menu when it is enabled

MODELS

 All

Access: RW

Name	Type
minimize	bool

ui.osd

Enable or disable on screen display

MODELS

 All

Access: RW

Name	Type
osd	bool

ui.patternmenu

Show or hide the pattern menu shortcut

MODELS

 All

Access: RW

Name	Type
patternmenu	bool

ui.poweroffhint

When true, a dialog shows info about powering down

MODELS

 All

Access: RW

Name	Type
poweroffhint	bool

ui.sensors.airpressure.unit

The unit.

MODELS

 All

Access: R

Name	Type
unit	string

ui.sensors.airpressure.value

The value.

MODELS

 All

Access: R

Name	Type
value	{ }
Name	Type
base	float
hints	{ }

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

ui.sensors.ambienttemperature.unit

The unit.

MODELS All

Access: R

Name	Type
unit	string

ui.sensors.ambienttemperature.value

The value.

MODELS All

Access: R

Name	Type
value	{ }
Name	Type
base	float
hints	{ }

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

ui.sensors.humidity.unit

The unit.

MODELS All

Access: R

Name	Type
unit	string

ui.sensors.humidity.value

The value.

MODELS All

Access: R

Name	Type
value	{ }
Name	Type
base	float
hints	{ }

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

ui.sensors.mainsvoltage.unit

The unit.

MODELS

 All

Access: R

Name	Type
unit	string

ui.sensors.mainsvoltage.value

The value.

MODELS

 All

Access: R

Name	Type
value	{ }
Name	Type
base	float
hints	{ }

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

ui.showmessage

Show a persistent message on the screen

MODELS

 All

Access: R

Name	Type
showmessage	bool

ui.sourcemenu

Show or hide the input source shortcut menu

MODELS

 All

Access: RW

Name	Type
sourcemenu	bool

ui.sourcesignal

Show/hide the source signal information popup

MODELS All

Access: RW

Name	Type
sourcesignal	bool

ui.sourcesignalposition

Placement of the source signal information

MODELS All

Access: RW

Name	Type
sourcesignalposition	enum
Values	
"TopLeft"	
"Top"	
"TopRight"	
"Right"	
"BottomRight"	
"Bottom"	
"BottomLeft"	
"Left"	
"Center"	

ui.splashscreen.show

Value to indicate if the splash screen should be shown

MODELS All

Access: R

Name	Type
show	bool

ui.splashscreen.timeoutseconds

Time (in seconds) the splash screen is shown on startup

MODELS All

Access: R

Name	Type
timeoutseconds	{ }

Name	Type
base	int
hints	{ }

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

ui.stealthmode

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

MODELS All

Access: RW

Name	Type
stealthmode	enum

Values
"Off"
"On"
"OnUntilReboot"

ui.temperature

Preferred unit for display of temperature values

MODELS All

Access: RW

Name	Type
temperature	enum

Values
"C"
"F"

ui.theme

The theme setting of the user interface.

MODELS All

Access: RW

Name	Type
theme	enum

Values
"Light"
"Dark"

ui.touchscreen.mode

Enables or disables the touchscreen test.

MODELS All

Access: R

Name	Type
mode	enum
Values	
"Normal"	
"Test"	

ui.touchscreen.status

Current status of the touchscreen text

MODELS All

Access: RW

Name	Type
status	enum
Values	
"Inactive"	
"Ongoing"	
"Complete"	

user.admin.activesessioncount

Description not provided

MODELS All

Access: R

Name	Type
activesessioncount	{ }
NameType	
base	int
hints	{ }
NameType	

user.admin.enabled

Description not provided

MODELS All

Access: R

Name	Type
enabled	bool

user.admin.group

Description not provided

MODELS

 All

Access: R

Name	Type
group	string

user.admin.publickey

Description not provided

MODELS

 All

Access: R

Name	Type
publickey	string

user.admin.username

Description not provided

MODELS

 All

Access: R

Name	Type
username	string

user.admin.userslug

Description not provided

MODELS

 All

Access: R

Name	Type
userslug	string

user.authenticationrequired

Require authentication of all users

MODELS

 All

Access: R

Name	Type
authenticationrequired	bool

user.availablegroups

The available user groups

MODELS

 All

Access: R

Name	Type
availablegroups	[string]

user.currentuser.activesessioncount

Description not provided

MODELS All

Access: R

Name	Type						
activesessioncount	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

user.currentuser.enabled

Description not provided

MODELS All

Access: R

Name	Type
enabled	bool

user.currentuser.group

Description not provided

MODELS All

Access: R

Name	Type
group	string

user.currentuser.publickey

Description not provided

MODELS All

Access: R

Name	Type
publickey	string

user.currentuser.username

Description not provided

MODELS All

Access: R

Name	Type
username	string

user.currentuser.userslug

Description not provided

MODELS All

Access: R

Name	Type
userslug	string

user.list

The list of all users

MODELS All

Access: R

Name	Type
list	[{ }]

{ }

Name	Type
Username	string
UserSlug	string
Group	string
Sessions	{ }

Name	Type
base	int
hints	{ }

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

user.pincodeauthenticationenabled

Allow authenticating users with pin code

MODELS All

Access: R

Name	Type
pincodeauthenticationenabled	bool

user.poweruser.activesessioncount

Description not provided

MODELS All

Access: R

Name	Type
activesessioncount	{ }

Name	Type
base	int
hints	{ }

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

user.poweruser.enabled

Description not provided

MODELS All

Access: R

Name	Type
enabled	bool

user.poweruser.group

Description not provided

MODELS All

Access: R

Name	Type
group	string

user.poweruser.publickey

Description not provided

MODELS All

Access: R

Name	Type
publickey	string

user.poweruser.username

Description not provided

MODELS All

Access: R

Name	Type
username	string

user.poweruser.userslug

Description not provided

MODELS

All

Access: R

Name	Type
userslug	string

user.resetadministratorchallenge

Challenge for resetting the administrator user

MODELS

All

Access: R

Name	Type
resetadministratorchallenge	{ }
Name	Type
base	int
hints	{ }
Name	Type

user.testuser.activesessioncount

Description not provided

MODELS

All

Access: R

Name	Type
activesessioncount	{ }
Name	Type
base	int
hints	{ }
Name	Type

user.testuser.enabled

Description not provided

MODELS

All

Access: R

Name	Type
enabled	bool

user.testuser.group

Description not provided

MODELS

All

Access: R

Name	Type
group	string

user.testuser.publickey

Description not provided

MODELS All

Access: R

Name	Type
publickey	string

user.testuser.username

Description not provided

MODELS All

Access: R

Name	Type
username	string

user.testuser.userslug

Description not provided

MODELS All

Access: R

Name	Type
userslug	string

user.user.activesessioncount

Description not provided

MODELS All

Access: R

Name	Type						
activesessioncount	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

user.user.enabled

Description not provided

MODELS All

Access: R

Name	Type
enabled	bool

user.user.group

Description not provided

MODELS All

Access: R

Name	Type
group	string

user.user.publickey

Description not provided

MODELS All

Access: R

Name	Type
publickey	string

user.user.username

Description not provided

MODELS All

Access: R

Name	Type
username	string

user.user.userslug

Description not provided

MODELS All

Access: R

Name	Type
userslug	string

Methods

Alphabetical list of all methods

.authenticate

[DEPRECATED] Authenticate using pin code

MODELS

 All

Parameters

Name	Type
code	int

Return value

Name	Type
userslug	string

.generatetoken

Request a new token for the currently authenticated user

MODELS

 All

This method does not require any parameters.

Return value

Name	Type
token	string

.getchallenge

Get the challenge for key-based authentication

MODELS

 All

This method does not require any parameters.

Return value

Name	Type
challenge	string

.introspect

Introspect the API for a specified object path

MODELS

 All

Parameters

Name	Type
object	string
recursive	bool
descriptions	bool

Return value

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

.login

Login using username and signed challenge (set token true to receive a token)

MODELS

All

Parameters

Name	Type
username	string
signature	string
token	bool

Return value

Name	Type
token	string

.logout

De-authenticate

MODELS

All

This method does not require any parameters.

This method has no return value.

dmx.listchannels

Return a list of available channel names

MODELS

All

This method does not require any parameters.

Return value

Name	Type
modes	[string]

dmx.listmodes

Return a list of all modes

MODELS

All

This method does not require any parameters.

Return value

Name	Type
modes	[string]

environment.getalarminfo

Description not provided

MODELS

All

This method does not require any parameters.

Return value

Name	Type
alarminfo	[{ }]

{ }

Name	Type
severity	string
timestamp	string
source	string
description	string

environment.getcontrolblocks

Description not provided

MODELS

All

Parameters

Name	Type
type	enum
	Values "Actuator" "Alarm" "Controller" "Filter" "GenericBlock" "Sensor"
valuetype	enum
	Values "ADC"

"Altitude"
"Angle"
"Any"
"Average"
"Comparison"
"Constant"
"Contamination"
"Coordinate"
"Current"
"Delay"
"Difference"
"Disabled"
"Driver"
"Formula"
"Humidity"
"Interpolation"
"Light"
"Limit"
"LiquidPressure"
"Manual"
"Median"
"Mode"
"Noise"
"PID"
"PWM"
"Peltier"
"Power"
"Pressure"
"Pump"
"Range"
"Ratio"
"Resistance"
"Simulation"
"Speed"
"State"
"Temperature"
"Threshold"
"Upgrader"
"Voltage"
"Volume"
"Waveform"
"Weighting"

Return value

Name	Type						
blocks	[{ }]						
	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>key</td><td>string</td></tr><tr><td>value</td><td>{ }</td></tr></table>	Name	Type	key	string	value	{ }
Name	Type						
key	string						
value	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

gpio.gpo.trigger1.listavailablemodes

List of possible GPO modes.

MODELS All

This method does not require any parameters.

Return value

Name	Type			
modes	[enum]			
	<table><tr><th>Values</th></tr><tr><td>"Level"</td></tr><tr><td>"PulseTrain"</td></tr></table>	Values	"Level"	"PulseTrain"
Values				
"Level"				
"PulseTrain"				

gpio.gpo.trigger1.pulse

Single toggle value of output

MODELS All

This method does not require any parameters.

This method has no return value.

gpio.gpo.trigger2.listavailablemodes

List of possible GPO modes.

MODELS All

This method does not require any parameters.

Return value

Name	Type
modes	[enum]

Values
"Level"
"PulseTrain"

gpio.gpo.trigger2.pulse

Single toggle value of output

MODELS All

This method does not require any parameters.

This method has no return value.

gpio.listinputs

List the names of all General Purpose Inputs

MODELS All

This method does not require any parameters.

Return value

Name	Type
arg0	[string]

gpio.listoutputs

List the names of all General Purpose Outputs

MODELS All

This method does not require any parameters.

Return value

Name	Type
arg0	[string]

illumination.clo.engage

Engage CLO at the current light level

MODELS All

This method does not require any parameters.

This method has no return value.

image.brilliantcolor.list

List possible BrilliantColor modes.

MODELS All

This method does not require any parameters.

Return value

Name	Type
modes	[string]

image.color.p7.custom.copypresettocustom

Description not provided

MODELS All

Parameters

Name	Type
presetname	string

This method has no return value.

image.color.p7.custom.makebalanced

Create a balanced colorspace from the current primaries

MODELS All

This method does not require any parameters.

This method has no return value.

image.color.p7.custom.resetpreset

Reset preset back to its default values

MODELS All

Parameters

Name	Type
presetname	string

This method has no return value.

image.color.p7.custom.resettonative

Description not provided

MODELS All

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.

MODELS

 All

This method does not require any parameters.

This method has no return value.

image.connector.displayport.edid.list

List system EDIDs available for this connector

MODELS

 All

This method does not require any parameters.

Return value

Name	Type						
selections	[{ }]						
	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>group</td><td>string</td></tr><tr><td>edids</td><td>[string]</td></tr></table>	Name	Type	group	string	edids	[string]
Name	Type						
group	string						
edids	[string]						

image.connector.hdmi.edid.list

List system EDIDs available for this connector

MODELS

 All

This method does not require any parameters.

Return value

Name	Type						
selections	[{ }]						
	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>group</td><td>string</td></tr><tr><td>edids</td><td>[string]</td></tr></table>	Name	Type	group	string	edids	[string]
Name	Type						
group	string						
edids	[string]						

image.connector.list

Description not provided

MODELS

 All

This method does not require any parameters.

Return value

Name	Type
connectors	[string]

image.display.listdisplaymodes

List possible display modes.

MODELS

 All

This method does not require any parameters.

Return value

Name	Type
displaymodes	[enum]
Values	
"Mono"	
"AutoStereo"	
"ActiveStereo"	
"NightVision"	
"IGPixelShift"	
"IGPixelShiftNV"	
"IGPixelShiftFullNV"	

image.list

List all available orientation modes

MODELS

 All

This method does not require any parameters.

Return value

Name	Type
orientationmode	[string]

image.listgammatypes

List all available gamma types.

MODELS

 All

This method does not require any parameters.

Return value

Name	Type
gammatypes	[string]

image.processing.blacklevel.basicblacklevel.getblacklevelarea

Returns the four boxes describing the black level edges.

MODELS

All

Parameters

Name	Type						
resolution_width	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
resolution_height	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

Return value

Name	Type						
output	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>Top</td><td>{ }</td></tr></table>	Name	Type	Top	{ }		
Name	Type						
Top	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>Start1</td><td>{ }</td></tr></table>	Name	Type	Start1	{ }		
Name	Type						
Start1	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>X</td><td>{ }</td></tr></table>	Name	Type	X	{ }		
Name	Type						
X	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>Y</td><td>{ }</td></tr></table>	Name	Type	Y	{ }		
Name	Type						
Y	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>Start2</td><td>{ }</td></tr></table>	Name	Type	Start2	{ }		
Name	Type						
Start2	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>X</td><td>{ }</td></tr></table>	Name	Type	X	{ }		
Name	Type						
X	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>Y</td><td>{ }</td></tr></table>	Name	Type	Y	{ }		
Name	Type						
Y	{ }						

																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														</
--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	----

```
hints { }
```

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

```
Start2 { }
```

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

$$X \quad \{ \}$$

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

```
base float
```

```
hints { }
```

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

$$Y \quad \{ \}$$

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

```
base float
```

```
hints { }
```

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

Right { }

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

```
Start1 { }
```

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

$$X \quad \{ \}$$

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

```
base float
```

```
hints { }
```

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

$$Y \quad \{ \}$$

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

```
base float
```

```
hints { }
```

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

```
Start2 { }
```

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

$$X \quad \{ \quad \}$$

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

```
base    float
```

```
hints { }
```

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

$$Y \quad \{ \}$$

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

```
base float
```

```
hints { }
```

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

image.processing.blacklevel.basicblacklevel.getwarpedblacklevelarea

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

MODELS

All

Parameters

Name	Type						
resolution_width	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
resolution_height	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

Return value

Name	Type						
output	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>Top</td><td>{ }</td></tr></table>	Name	Type	Top	{ }		
Name	Type						
Top	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>Start1</td><td>{ }</td></tr></table>	Name	Type	Start1	{ }		
Name	Type						
Start1	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>X</td><td>{ }</td></tr></table>	Name	Type	X	{ }		
Name	Type						
X	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>Y</td><td>{ }</td></tr></table>	Name	Type	Y	{ }		
Name	Type						
Y	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>Start2</td><td>{ }</td></tr></table>	Name	Type	Start2	{ }		
Name	Type						
Start2	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>X</td><td>{ }</td></tr></table>	Name	Type	X	{ }		
Name	Type						
X	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>Y</td><td>{ }</td></tr></table>	Name	Type	Y	{ }		
Name	Type						
Y	{ }						

			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type								
base	float								
hints	{ }								
			<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type								
Bottom	{ }								
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>Start1</td><td>{ }</td></tr></table>	Name	Type	Start1	{ }				
Name	Type								
Start1	{ }								
		<table><tr><th>Name</th><th>Type</th></tr><tr><td>X</td><td>{ }</td></tr></table>	Name	Type	X	{ }			
Name	Type								
X	{ }								
			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type								
base	float								
hints	{ }								
			<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type								
		Y	{ }						
			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type								
base	float								
hints	{ }								
			<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type								
	Start2	{ }							
		<table><tr><th>Name</th><th>Type</th></tr><tr><td>X</td><td>{ }</td></tr></table>	Name	Type	X	{ }			
Name	Type								
X	{ }								
			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type								
base	float								
hints	{ }								
			<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type								
		Y	{ }						
			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type								
base	float								
hints	{ }								
			<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type								
Left	{ }								
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>Start1</td><td>{ }</td></tr></table>	Name	Type	Start1	{ }				
Name	Type								
Start1	{ }								
		<table><tr><th>Name</th><th>Type</th></tr><tr><td>X</td><td>{ }</td></tr></table>	Name	Type	X	{ }			
Name	Type								
X	{ }								
			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type								
base	float								
hints	{ }								
			<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type								
		Y	{ }						
			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr></table>	Name	Type	base	float		
Name	Type								
base	float								

		hints	{ }		
				<u>Name</u>	<u>Type</u>
Start2		{ }			
		<u>Name</u>	<u>Type</u>		
		X	{ }		
				<u>Name</u>	<u>Type</u>
				base	float
				hints	{ }
				<u>Name</u>	<u>Type</u>
		Y	{ }		
		<u>Name</u>	<u>Type</u>		
				base	float
				hints	{ }
				<u>Name</u>	<u>Type</u>
Right	{ }				
		<u>Name</u>	<u>Type</u>		
		Start1	{ }		
		<u>Name</u>	<u>Type</u>		
		X	{ }		
				<u>Name</u>	<u>Type</u>
				base	float
				hints	{ }
				<u>Name</u>	<u>Type</u>
		Y	{ }		
		<u>Name</u>	<u>Type</u>		
				base	float
				hints	{ }
				<u>Name</u>	<u>Type</u>
		Start2	{ }		
		<u>Name</u>	<u>Type</u>		
		X	{ }		
				<u>Name</u>	<u>Type</u>
				base	float
				hints	{ }
				<u>Name</u>	<u>Type</u>
		Y	{ }		
		<u>Name</u>	<u>Type</u>		
				base	float
				hints	{ }
				<u>Name</u>	<u>Type</u>

image.processing.blacklevel.file.delete

Deletes a file with the given name.

MODELS

 All

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

MODELS

 All

This method does not require any parameters.

Return value

Name	Type
filenames	[string]

image.processing.blend.basicblend.getblendarea

Returns the four boxes describing the blend edges.

MODELS

 All

Parameters

Name	Type						
resolution_width	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
resolution_height	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

Return value

Name	Type								
output	{ }								
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>Top</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>	Name	Type	Top	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type
Name	Type								
Top	{ }								
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type						
Name	Type								

Start1 { }

Name	Type
X	{ }

Name	Type
base	float
hints	{ }

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

Y { }

Name	Type
base	float
hints	{ }

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

Start2 { }

Name	Type
X	{ }

Name	Type
base	float
hints	{ }

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

Y { }

Name	Type
base	float
hints	{ }

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

Width1 { }

Name	Type
X	{ }

Name	Type
base	float
hints	{ }

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

Y { }

Name	Type
base	float
hints	{ }

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

Width2 { }

Name	Type
X	{ }

Name	Type
base	float
hints	{ }

<

Width2 { }	hints { }	
Left { }		
Start2 { }		
Width1 { }		

Width2	{ }	X		
			Name	Type
			base	float
			hints	{ }
		Y		
			Name	Type
			base	float
			hints	{ }
			Name	Type
Right	{ }	X		
			Name	Type
			base	float
			hints	{ }
		Y		
			Name	Type
			base	float
			hints	{ }
			Name	Type
Start1	{ }	X		
			Name	Type
			base	float
			hints	{ }
		Y		
			Name	Type
			base	float
			hints	{ }
			Name	Type
Start2	{ }	X		
			Name	Type
			base	float
			hints	{ }
			Name	Type

					Name Type
			Y	{ }	
					Name Type
					base float
					hints { }
					Name Type
		Width1	{ }		
					Name Type
			X	{ }	
					Name Type
					base float
					hints { }
					Name Type
			Y	{ }	
					Name Type
					base float
					hints { }
					Name Type
		Width2	{ }		
					Name Type
			X	{ }	
					Name Type
					base float
					hints { }
					Name Type
			Y	{ }	
					Name Type
					base float
					hints { }
					Name Type

image.processing.blend.basicblend.getwarpedblendarea

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

MODELS All

Parameters

Name	Type
resolution_width	{ }
	Name Type
	base float
	hints { }
	Name Type

resolution_height { }

Name	Type
base	float
hints	{ }

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

Return value

Name	Type
output	{ }

Name	Type
Top	{ }

Name	Type
Start1	{ }

Name	Type
X	{ }

Name	Type
base	float
hints	{ }

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

Y	{ }
---	-----

Name	Type
base	float
hints	{ }

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

Start2	{ }
--------	-----

Name	Type
X	{ }

Name	Type
base	float
hints	{ }

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

Y	{ }
---	-----

Name	Type
base	float
hints	{ }

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

Width1	{ }
--------	-----

Name	Type
X	{ }

Name	Type
base	float
hints	{ }

```

graph TD
    Program["Program { }"] --> Width1["Width1 { }"]
    Program --> Width2["Width2 { }"]
    Program --> Bottom["Bottom { }"]
    Width1 --> X1["X { }"]
    Width1 --> Y1["Y { }"]
    Width2 --> X2["X { }"]
    Width2 --> Y2["Y { }"]
    Bottom --> X3["X { }"]
    Bottom --> Y3["Y { }"]
    X1 --> base1["base float"]
    X1 --> hints1["hints { }"]
    Y1 --> base2["base float"]
    Y1 --> hints2["hints { }"]
    X2 --> base3["base float"]
    X2 --> hints3["hints { }"]
    Y2 --> base4["base float"]
    Y2 --> hints4["hints { }"]
    X3 --> base5["base float"]
    X3 --> hints5["hints { }"]
    Y3 --> base6["base float"]
    Y3 --> hints6["hints { }"]
  
```

Width1 { }	hints { }	
	NameType	
	NameType	
	X { }	
	NameType	
	base float	
	hints { }	
	NameType	
	Y { }	
	NameType	
Width2 { }	NameType	
	X { }	
	NameType	
	base float	
	hints { }	
	NameType	
	Y { }	
	NameType	
	base float	
	hints { }	
Left { }	NameType	
	Start1 { }	
	NameType	
	X { }	
	NameType	
	base float	
	hints { }	
	NameType	
	Y { }	
	NameType	
Start2 { }	NameType	
	NameType	
	NameType	

X

{ }

Name

Type

base

float

hints

{ }

Name

Type

Y

{ }

Name

Type

base

float

hints

{ }

Name

Type

Width1

{ }

Name

Type

X

{ }

Name

Type

base

float

hints

{ }

Name

Type

Y

{ }

Name

Type

base

float

hints

{ }

Name

Type

Width2

{ }

Name

Type

X

{ }

Name

Type

base

float

hints

{ }

Name

Type

Y

{ }

Name

Type

base

float

hints

{ }

Name

Type

Right

{ }

Name

Type

Start1

{ }

Name

Type

X

{ }

Name

Type

base

float

hints

{ }

				Name	Type
		Y	{ }		
				Name	Type
				base	float
				hints	{ }
				Name	Type
Start2	{ }				
				Name	Type
		X	{ }		
				Name	Type
				base	float
				hints	{ }
				Name	Type
		Y	{ }		
				Name	Type
				base	float
				hints	{ }
				Name	Type
Width1	{ }				
				Name	Type
		X	{ }		
				Name	Type
				base	float
				hints	{ }
				Name	Type
		Y	{ }		
				Name	Type
				base	float
				hints	{ }
				Name	Type
Width2	{ }				
				Name	Type
		X	{ }		
				Name	Type
				base	float
				hints	{ }
				Name	Type
		Y	{ }		
				Name	Type
				base	float
				hints	{ }
				Name	Type

image.processing.blend.file.delete

Deletes a file with the given name.

MODELS All

Parameters

Name	Type
filename	string

This method has no return value.

image.processing.blend.file.list

Returns a list of available blend files

MODELS All

This method does not require any parameters.

Return value

Name	Type
filenames	[string]

image.processing.warp.file.delete

Deletes a file with the given name.

MODELS All

Parameters

Name	Type
filename	string

This method has no return value.

image.processing.warp.file.list

Returns a list of available warp files

MODELS All

This method does not require any parameters.

Return value

Name	Type
filenames	[string]

image.processing.warp.fourcorners.getscaledcorners

Get the corners scaled to the given resolution

Parameters

Name	Type						
resolution	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>		Name	Type	x	{ }		
Name	Type						
x	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
y	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

Return value

Name	Type						
corners	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>TopLeft</td><td>{ }</td></tr></table>		Name	Type	TopLeft	{ }		
Name	Type						
TopLeft	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>		Name	Type	x	{ }		
Name	Type						
x	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
y	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						
TopRight	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>		Name	Type	x	{ }		
Name	Type						
x	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						
y	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

The diagram illustrates the construction of a bottom-left corner from a bottom-right corner. It consists of two main parts, each containing a table of variables and their types, and a table of hints.

Initial Bottom-Right Corner:

Name	Type
x	{ }

Below this table is a table of hints:

base	int
hints	{ }

Resulting Bottom-Left Corner:

After a 90-degree counter-clockwise rotation, the variables are rearranged:

Name	Type
y	{ }

Below this table is a table of hints:

base	int
hints	{ }

image.processing.warp.warpscaledpoints

Takes an array of points and returns their warped equivalents.

MODELS All

Parameters

Name	Type
points	[{ }]
	{ }
X	{ }

Name	Type
base	float
hints	{ }

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

Y	{ }			
			Name	Type
			base	float
resolution	{ }		hints	{ }
			Name	Type
X	{ }			
			Name	Type
			base	float
Y	{ }		hints	{ }
			Name	Type
Y	{ }			
			Name	Type
			base	float
Y	{ }		hints	{ }
			Name	Type

Return value

Name	Type						
points	[{ }]						
	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>X</td><td>{ }</td></tr></table>	Name	Type	X	{ }		
Name	Type						
X	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>Y</td><td>{ }</td></tr></table>	Name	Type	Y	{ }		
Name	Type						
Y	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

image.processing.warpgrid.getgrid

Get the current grid points as normalized and relative

MODELS All

This method does not require any parameters.

Return value

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

grid [{ }]							
{ }							
<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>		Name	Type	x	{ }		
Name	Type						
x	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr></table>		Name	Type	y	{ }		
Name	Type						
y	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	float	hints	{ }
Name	Type						
base	float						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

image.processing.warpgrid.getgridsize

Description not provided

MODELS All

This method does not require any parameters.

Return value

Name	Type						
size	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>		Name	Type	x	{ }		
Name	Type						
x	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>y</td><td>{ }</td></tr></table>		Name	Type	y	{ }		
Name	Type						
y	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

image.processing.warpgrid.getscaledgrid

Get the current grid scaled to the given resolution

MODELS All

Parameters

Name	Type
resolution	{ }

Name	Type
x	{ }

Name	Type
base	int
hints	{ }

Name	Type
y	{ }

Name	Type
base	int
hints	{ }

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

Return value

Name	Type
grid	[{ }]

{ }

Name	Type
x	{ }

Name	Type
base	float
hints	{ }

Name	Type
y	{ }

Name	Type
base	float
hints	{ }

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

image.resolution.list

List all possible resolutions.

MODELS All

This method does not require any parameters.

Return value

Name	Type
resolutions	[string]

image.source.displayport.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

Name	Type						
connectors	[{ }]						
	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>gridposition</td><td>{ }</td></tr></table>	Name	Type	gridposition	{ }		
Name	Type						
gridposition	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>row</td><td>{ }</td></tr></table>	Name	Type	row	{ }		
Name	Type						
row	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>column</td><td>{ }</td></tr></table>	Name	Type	column	{ }		
Name	Type						
column	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>view</td><td>{ }</td></tr></table>	Name	Type	view	{ }		
Name	Type						
view	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>plane</td><td>{ }</td></tr></table>	Name	Type	plane	{ }		
Name	Type						
plane	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>name</td><td>string</td></tr></table>	Name	Type	name	string		
Name	Type						
name	string						

image.source.hdmi.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

Name	Type		
connectors	[{ }]		
	{ }		
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type
Name	Type		

gridposition { }																																																															
<table><tr><th>Name</th><th>Type</th></tr><tr><td>row</td><td>{ }</td></tr></table>		Name	Type	row	{ }																																																										
Name	Type																																																														
row	{ }																																																														
<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">column { }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">view { }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">plane { }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">name string</td></tr></table></td></tr></table></td></tr></table></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type	column { }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">view { }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">plane { }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">name string</td></tr></table></td></tr></table></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type	view { }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">plane { }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">name string</td></tr></table></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type	plane { }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">name string</td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type	name string	
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																																																								
Name	Type																																																														
base	int																																																														
hints	{ }																																																														
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type																																																												
Name	Type																																																														
column { }																																																															
<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">view { }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">plane { }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">name string</td></tr></table></td></tr></table></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type	view { }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">plane { }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">name string</td></tr></table></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type	plane { }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">name string</td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type	name string																	
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																																																								
Name	Type																																																														
base	int																																																														
hints	{ }																																																														
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type																																																												
Name	Type																																																														
view { }																																																															
<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">plane { }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">name string</td></tr></table></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type	plane { }		<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">name string</td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type	name string																																	
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																																																								
Name	Type																																																														
base	int																																																														
hints	{ }																																																														
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type																																																												
Name	Type																																																														
plane { }																																																															
<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td colspan="2"><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td colspan="2">name string</td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type	name string																																																	
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																																																								
Name	Type																																																														
base	int																																																														
hints	{ }																																																														
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type																																																												
Name	Type																																																														
name string																																																															

image.source.list

List all available sources

MODELS All

This method does not require any parameters.

Return value

Name	Type
sources	[string]

image.stereo.listdarktime

List all possible darktime values (in us).

MODELS All

This method does not require any parameters.

Return value

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

darktime	
[{ }]	
{ }	
Name	Type
base	int
hints	{ }
Name	Type

image.testpattern.file.delete

Deletes a file with the given name.

MODELS

All

Parameters

Name	Type
filename	string

This method has no return value.

image.testpattern.file.list

Get a list of available custom uploaded patterns

MODELS

All

This method does not require any parameters.

Return value

Name	Type
patterns	[string]

image.testpattern.getProperties

Get the properties of a pattern

MODELS

All

Parameters

Name	Type
id	string

Return value

Name	Type
properties	[{ }]
{ }	
Name	Type
key	string
value	string

image.testpattern.iterator.calibration.listpossiblepatterns

Return the list of possible testpatterns to iterate.

MODELS

 All

This method does not require any parameters.

Return value

Name	Type
list	[string]

image.testpattern.iterator.calibration.next

Show the next testpattern in the list.

MODELS

 All

This method does not require any parameters.

This method has no return value.

image.testpattern.iterator.iris.listpossiblepatterns

Return the list of possible testpatterns to iterate.

MODELS

 All

This method does not require any parameters.

Return value

Name	Type
list	[string]

image.testpattern.iterator.iris.next

Show the next testpattern in the list.

MODELS

 All

This method does not require any parameters.

This method has no return value.

image.testpattern.iterator.optics.listpossiblepatterns

Return the list of possible testpatterns to iterate.

MODELS

 All

This method does not require any parameters.

Return value

Name	Type
list	[string]

image.testpattern.iterator.optics.next

Show the next testpattern in the list.

MODELS All

This method does not require any parameters.

This method has no return value.

image.testpattern.iterator.shift.listpossiblepatterns

Return the list of possible testpatterns to iterate.

MODELS All

This method does not require any parameters.

Return value

Name	Type
list	[string]

image.testpattern.iterator.shift.next

Show the next testpattern in the list.

MODELS All

This method does not require any parameters.

This method has no return value.

image.testpattern.iterator.stereo.listpossiblepatterns

Return the list of possible testpatterns to iterate.

MODELS All

This method does not require any parameters.

Return value

Name	Type
list	[string]

image.testpattern.iterator.stereo.next

Show the next testpattern in the list.

MODELS

All

This method does not require any parameters.

This method has no return value.

image.testpattern.iterator.xpr.listpossiblepatterns

Return the list of possible testpatterns to iterate.

MODELS

All

This method does not require any parameters.

Return value

Name	Type
list	[string]

image.testpattern.iterator.xpr.next

Show the next testpattern in the list.

MODELS

All

This method does not require any parameters.

This method has no return value.

image.testpattern.list

Get a list of available patterns

MODELS

All

This method does not require any parameters.

Return value

Name	Type														
patterns	[{ }]														
	{ }														
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>name</td><td>string</td></tr><tr><td>location</td><td>string</td></tr><tr><td>id</td><td>string</td></tr><tr><td>above</td><td>bool</td></tr><tr><td>internal</td><td>bool</td></tr><tr><td>properties</td><td>[{ }]</td></tr></table>	Name	Type	name	string	location	string	id	string	above	bool	internal	bool	properties	[{ }]
Name	Type														
name	string														
location	string														
id	string														
above	bool														
internal	bool														
properties	[{ }]														
	{ }														
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type												
Name	Type														

key	string
value	string

image.testpattern.listinternal

Get a list of available internal patterns

MODELS All

This method does not require any parameters.

Return value

Name	Type														
patterns	[{ }]														
	{ }														
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>name</td><td>string</td></tr><tr><td>location</td><td>string</td></tr><tr><td>id</td><td>string</td></tr><tr><td>above</td><td>bool</td></tr><tr><td>internal</td><td>bool</td></tr><tr><td>properties</td><td>[{ }]</td></tr></table>	Name	Type	name	string	location	string	id	string	above	bool	internal	bool	properties	[{ }]
Name	Type														
name	string														
location	string														
id	string														
above	bool														
internal	bool														
properties	[{ }]														
	{ }														
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>key</td><td>string</td></tr><tr><td>value</td><td>string</td></tr></table>	Name	Type	key	string	value	string								
Name	Type														
key	string														
value	string														

image.testpattern.nosignal.list

Get a list of available NoSignal patterns

MODELS All

This method does not require any parameters.

Return value

Name	Type						
patterns	[{ }]						
	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>name</td><td>string</td></tr><tr><td>id</td><td>string</td></tr></table>	Name	Type	name	string	id	string
Name	Type						
name	string						
id	string						

image.testpattern.setproperties

Set the properties of a pattern

MODELS All

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_OK"

```
"KP_RIGHT"
"KP_DOWN"
"KP_MENU"
"KP_POWER"
"KP_BACK"
"KP_OSD"
"KP_LENS"
"KP_PATTERN"
"KP_SHUTTER"
"KP_INPUT"
"KP_STANDBY"
"TP_TOUCH"
"AP_POWER_OFF"
```

This method has no return value.

led.activity

Activates the LEDS when enabled and restarts the LED timeout counter

MODELS

 All

This method does not require any parameters.

This method has no return value.

led.list

Description not provided

MODELS

 All

This method does not require any parameters.

Return value

Name	Type
leds	[string]

led.muteall

Mutes the LEDS.

MODELS

 All

This method does not require any parameters.

This method has no return value.

lightmeasurement.getlightoutput

Description not provided

MODELS

All

This method does not require any parameters.

Return value

Name	Type						
lumens	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

network.list

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

MODELS

All

This method does not require any parameters.

Return value

Name	Type
devices	[string]

notification.caution.list

Retrieve all active notifications for the given severity

MODELS

All

This method does not require any parameters.

Return value

Name	Type															
notifications	[{ }]															
	{ }															
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>severity</td><td>enum</td></tr><tr><td></td><td><table><tr><th>Values</th></tr><tr><td>"INFO"</td></tr><tr><td>"CAUTION"</td></tr><tr><td>"WARNING"</td></tr><tr><td>"ERROR"</td></tr></table></td></tr><tr><td>id</td><td>string</td></tr><tr><td>code</td><td>string</td></tr></table>	Name	Type	severity	enum		<table><tr><th>Values</th></tr><tr><td>"INFO"</td></tr><tr><td>"CAUTION"</td></tr><tr><td>"WARNING"</td></tr><tr><td>"ERROR"</td></tr></table>	Values	"INFO"	"CAUTION"	"WARNING"	"ERROR"	id	string	code	string
Name	Type															
severity	enum															
	<table><tr><th>Values</th></tr><tr><td>"INFO"</td></tr><tr><td>"CAUTION"</td></tr><tr><td>"WARNING"</td></tr><tr><td>"ERROR"</td></tr></table>	Values	"INFO"	"CAUTION"	"WARNING"	"ERROR"										
Values																
"INFO"																
"CAUTION"																
"WARNING"																
"ERROR"																
id	string															
code	string															

timestamp	string
message	string
timeout	string
visibility	string
doc	string
actions	[enum]

Values

"NONE "

"OK "

"CANCEL "

"YES "

"NO "

"SUPPRESS "

notification.dismiss

Dismiss the notification with the specified id

MODELS

All

Parameters

Name	Type
id	string
response	enum

Values

"NONE "

"OK "

"CANCEL "

"YES "

"NO "

"SUPPRESS "

This method has no return value.

notification.error.list

Retrieve all active notifications for the given severity

MODELS

All

This method does not require any parameters.

Return value

Name	Type
notifications	[{ }]

{ }

Name	Type
severity	enum
Values	
"INFO"	
"CAUTION"	
"WARNING"	
"ERROR"	
id	string
code	string
timestamp	string
message	string
timeout	string
visibility	string
doc	string
actions	[enum]
Values	
"NONE"	
"OK"	
"CANCEL"	
"YES"	
"NO"	
"SUPPRESS"	

notification.info.list

Retrieve all active notifications for the given severity

MODELS All

This method does not require any parameters.

Return value

Name	Type																		
notifications	[{ }]																		
{ }																			
<table><tr><th>Name</th><th>Type</th></tr><tr><td>severity</td><td>enum</td></tr><tr><td colspan="2"> Values </td></tr><tr><td colspan="2">"INFO"</td></tr><tr><td colspan="2">"CAUTION"</td></tr><tr><td colspan="2">"WARNING"</td></tr><tr><td colspan="2">"ERROR"</td></tr><tr><td>id</td><td>string</td></tr><tr><td>code</td><td>string</td></tr></table>		Name	Type	severity	enum	Values		"INFO"		"CAUTION"		"WARNING"		"ERROR"		id	string	code	string
Name	Type																		
severity	enum																		
Values																			
"INFO"																			
"CAUTION"																			
"WARNING"																			
"ERROR"																			
id	string																		
code	string																		

timestamp	string
message	string
timeout	string
visibility	string
doc	string
actions	[enum]

Values

"NONE"

"OK"

"CANCEL"

"YES"

"NO"

"SUPPRESS"

notification.list

List all active notifications

MODELS

All

This method does not require any parameters.

Return value

Name	Type
notifications	[{ }]

{ }

Name	Type
severity	enum

Values

"INFO"

"CAUTION"

"WARNING"

"ERROR"

id	string
code	string
timestamp	string
arguments	[string]
message	string
timeout	string
visibility	string
doc	string
actions	[enum]

Values

"NONE"

"OK"

"CANCEL "

"YES "

"NO "

"SUPPRESS"

messageTemplate string

notification.listsuppressed

Get a list of suppressed notification codes

MODELS All

This method does not require any parameters.

Return value

Name	Type
list	[string]

notification.log

List received notifications

MODELS All

Parameters

Name	Type						
minimumseverity	enum						
	<table><tr><th>Values</th></tr><tr><td>"INFO"</td></tr><tr><td>"CAUTION"</td></tr><tr><td>"WARNING"</td></tr><tr><td>"ERROR"</td></tr></table>	Values	"INFO"	"CAUTION"	"WARNING"	"ERROR"	
Values							
"INFO"							
"CAUTION"							
"WARNING"							
"ERROR"							
start	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
count	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

Return value

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

notifications [{ }] { }		
Name		Type
severity		enum
		Values "INFO" "CAUTION" "WARNING" "ERROR"
id		string
code		string
timestamp		string
arguments		[string]
message		string
timeout		string
visibility		string
doc		string
actions		[enum]
		Values "NONE" "OK" "CANCEL " "YES" "NO" "SUPPRESS"
messageTemplate		string

notification.suppress

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

MODELS All

Parameters

Name	Type
code	string

This method has no return value.

notification.unsuppress

No longer suppress a certain notification code

MODELS All

Parameters

Name	Type
code	string

This method has no return value.

notification.unsuppressall

No longer suppress any notification codes

MODELS All

This method does not require any parameters.

This method has no return value.

notification.warning.list

Retrieve all active notifications for the given severity

MODELS All

This method does not require any parameters.

Return value

Name	Type																																		
notifications	[{ }]																																		
	{ }																																		
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>severity</td><td>enum</td></tr><tr><td></td><td><table><tr><th>Values</th></tr><tr><td>"INFO"</td></tr><tr><td>"CAUTION"</td></tr><tr><td>"WARNING"</td></tr><tr><td>"ERROR"</td></tr></table></td></tr><tr><td>id</td><td>string</td></tr><tr><td>code</td><td>string</td></tr><tr><td>timestamp</td><td>string</td></tr><tr><td>message</td><td>string</td></tr><tr><td>timeout</td><td>string</td></tr><tr><td>visibility</td><td>string</td></tr><tr><td>doc</td><td>string</td></tr><tr><td>actions</td><td>[enum]</td></tr><tr><td></td><td><table><tr><th>Values</th></tr><tr><td>"NONE"</td></tr><tr><td>"OK"</td></tr><tr><td>"CANCEL "</td></tr><tr><td>"YES"</td></tr></table></td></tr></table>	Name	Type	severity	enum		<table><tr><th>Values</th></tr><tr><td>"INFO"</td></tr><tr><td>"CAUTION"</td></tr><tr><td>"WARNING"</td></tr><tr><td>"ERROR"</td></tr></table>	Values	"INFO"	"CAUTION"	"WARNING"	"ERROR"	id	string	code	string	timestamp	string	message	string	timeout	string	visibility	string	doc	string	actions	[enum]		<table><tr><th>Values</th></tr><tr><td>"NONE"</td></tr><tr><td>"OK"</td></tr><tr><td>"CANCEL "</td></tr><tr><td>"YES"</td></tr></table>	Values	"NONE"	"OK"	"CANCEL "	"YES"
Name	Type																																		
severity	enum																																		
	<table><tr><th>Values</th></tr><tr><td>"INFO"</td></tr><tr><td>"CAUTION"</td></tr><tr><td>"WARNING"</td></tr><tr><td>"ERROR"</td></tr></table>	Values	"INFO"	"CAUTION"	"WARNING"	"ERROR"																													
Values																																			
"INFO"																																			
"CAUTION"																																			
"WARNING"																																			
"ERROR"																																			
id	string																																		
code	string																																		
timestamp	string																																		
message	string																																		
timeout	string																																		
visibility	string																																		
doc	string																																		
actions	[enum]																																		
	<table><tr><th>Values</th></tr><tr><td>"NONE"</td></tr><tr><td>"OK"</td></tr><tr><td>"CANCEL "</td></tr><tr><td>"YES"</td></tr></table>	Values	"NONE"	"OK"	"CANCEL "	"YES"																													
Values																																			
"NONE"																																			
"OK"																																			
"CANCEL "																																			
"YES"																																			

"NO"

"SUPPRESS"

optics.focus.calibrate

Calibrate motor

MODELS All

This method does not require any parameters.

This method has no return value.

optics.focus.runforward

Run forward

MODELS All

This method does not require any parameters.

This method has no return value.

optics.focus.runforwardtime

Run forward for X milliseconds

MODELS All

Parameters

Name	Type
milliseconds	{ }

Name	Type
base	int
hints	{ }

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

This method has no return value.

optics.focus.runreverse

Run reverse

MODELS All

This method does not require any parameters.

This method has no return value.

optics.focus.runreversetime

Run reverse for X milliseconds

MODELS

All

Parameters

Name	Type
milliseconds	{ }

Name	Type
base	int
hints	{ }

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

This method has no return value.

optics.focus.stepforward

Step forward

MODELS

All

Parameters

Name	Type
steps	{ }

Name	Type
base	int
hints	{ }

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

This method has no return value.

optics.focus.stepreverse

Step reverse

MODELS

All

Parameters

Name	Type
steps	{ }

Name	Type
base	int
hints	{ }

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

This method has no return value.

optics.focus.stop

Stop

MODELS All

This method does not require any parameters.

This method has no return value.

optics.lens.list

List available lenses

MODELS All

This method does not require any parameters.

Return value

Name	Type
availablelenses	[{ }]
	{ }

Name	Type
Id	string
Description	string

optics.lensshift.horizontal.calibrate

Calibrate motor

MODELS All

This method does not require any parameters.

This method has no return value.

optics.lensshift.horizontal.runforward

Run forward

MODELS All

This method does not require any parameters.

This method has no return value.

optics.lensshift.horizontal.runforwardtime

Run forward for X milliseconds

MODELS All

Parameters

Name	Type
milliseconds	{ }

Name	Type
base	int
hints	{ }

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

This method has no return value.

optics.lensshift.horizontal.runreverse

Run reverse

MODELS All

This method does not require any parameters.

This method has no return value.

optics.lensshift.horizontal.runreversetime

Run reverse for X milliseconds

MODELS All

Parameters

Name	Type
milliseconds	{ }

Name	Type
base	int
hints	{ }

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

This method has no return value.

optics.lensshift.horizontal.stepforward

Step forward

MODELS All

Parameters

Name	Type
steps	{ }

Name	Type
base	int
hints	{ }

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

This method has no return value.

optics.lensshift.horizontal.stepreverse

Step reverse

MODELS All

Parameters

Name	Type						
steps	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

This method has no return value.

optics.lensshift.horizontal.stop

Stop

MODELS All

This method does not require any parameters.

This method has no return value.

optics.lensshift.vertical.calibrate

Calibrate motor

MODELS All

This method does not require any parameters.

This method has no return value.

optics.lensshift.vertical.runforward

Run forward

MODELS All

This method does not require any parameters.

This method has no return value.

optics.lensshift.vertical.runforwardtime

Run forward for X milliseconds

MODELS All

Parameters

Name	Type
milliseconds	{ }
Name	Type
base	int
hints	{ }
Name	Type

This method has no return value.

optics.lensshift.vertical.runreverse

Run reverse

MODELS All

This method does not require any parameters.

This method has no return value.

optics.lensshift.vertical.runreversetime

Run reverse for X milliseconds

MODELS All

Parameters

Name	Type
milliseconds	{ }
Name	Type
base	int
hints	{ }
Name	Type

This method has no return value.

optics.lensshift.vertical.stepforward

Step forward

MODELS All

Parameters

Name	Type
steps	{ }
Name	Type
base	int

hints { }

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

This method has no return value.

optics.lensshift.vertical.stepreverse

Step reverse

MODELS All

Parameters

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

steps { }

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

base int

hints { }

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

This method has no return value.

optics.lensshift.vertical.stop

Stop

MODELS All

This method does not require any parameters.

This method has no return value.

optics.shifttocenter

Shift lens to center of allowed shift range

MODELS All

This method does not require any parameters.

This method has no return value.

optics.shutter.getobjectpath

Get object path of shutter

MODELS All

This method does not require any parameters.

Return value

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

path string

optics.shutter.getstate

Get state of shutter

MODELS

 All

This method does not require any parameters.

Return value

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

optics.shutter.toggle

Toggle shutter position

MODELS

 All

This method does not require any parameters.

This method has no return value.

optics.zoom.calibrate

Calibrate motor

MODELS

 All

This method does not require any parameters.

This method has no return value.

optics.zoom.runforward

Run forward

MODELS

 All

This method does not require any parameters.

This method has no return value.

optics.zoom.runforwardtime

Run forward for X milliseconds

MODELS

 All

Parameters

Name	Type						
milliseconds	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

This method has no return value.

optics.zoom.runreverse

Run reverse

MODELS All

This method does not require any parameters.

This method has no return value.

optics.zoom.runreversetime

Run reverse for X milliseconds

MODELS All

Parameters

Name	Type						
milliseconds	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

This method has no return value.

optics.zoom.stepforward

Step forward

MODELS All

Parameters

Name	Type				
steps	{ }				
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr></table>	Name	Type	base	int
Name	Type				
base	int				

hints { }

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

This method has no return value.

optics.zoom.stepreverse

Step reverse

MODELS All

Parameters

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

steps { }

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

base int

hints { }

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

This method has no return value.

optics.zoom.stop

Stop

MODELS All

This method does not require any parameters.

This method has no return value.

peripheral.autodetect

Trigger auto detection of connected peripherals.

MODELS All

This method does not require any parameters.

This method has no return value.

peripheral.list

List alignment tools.

MODELS All

This method does not require any parameters.

Return value

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

modes

[string]

profile.activatepreset

Activate the profile associated with the given preset.

MODELS

All

Parameters

Name	Type
preset	{ }
Name	Type
base	int
hints	{ }
Name	Type

Return value

Name	Type
result	bool

profile.activateprofile

Activate the given profile

MODELS

All

Parameters

Name	Type
name	string

Return value

Name	Type
result	bool

profile.assignpreset

Assign a profile to a preset.

MODELS

All

Parameters

Name	Type
preset	{ }
Name	Type
base	int
hints	{ }
Name	Type

profile

string

Return value

Name	Type
result	bool

profile.createprofile

Create a profile containing the current values of the listed domains.

MODELS All

Parameters

Name	Type
profilename	string
domains	[string]

Return value

Name	Type
result	bool

profile.deleteprofile

Delete the given profile

MODELS All

Parameters

Name	Type
name	string

Return value

Name	Type
result	bool

profile.getdomainsforprofile

Get the list of domains stored in a profile.

MODELS All

Parameters

Name	Type
name	string

Return value

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

domains [string]

profile.presetforprofile

Get the preset assignment for given profile. -1 means unassigned.

MODELS

All

Parameters

Name	Type
name	string

Return value

Name	Type						
result	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

profile.profileforpreset

Get the profile associated with the given preset assignment. Empty string means unassigned.

MODELS

All

Parameters

Name	Type						
preset	{ }						
<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>		Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
<table><tr><th>Name</th><th>Type</th></tr></table>		Name	Type				
Name	Type						

Return value

Name	Type
result	string

remotecontrol.listsensors

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

MODELS

All

This method does not require any parameters.

Return value

Name	Type
sensors	[string]

scheduler.create

Create a new action for scheduler.

MODELS All

Parameters

Name	Type
name	string

Return value

Name	Type
path	string

statistics.illumination on uptime.getname

Name of the counter

MODELS All

This method does not require any parameters.

Return value

Name	Type
countername	string

statistics.illumination on uptime.getunit

Unit of measurements

MODELS All

This method does not require any parameters.

Return value

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

statistics.laser runtime.getname

Name of the counter

MODELS All

This method does not require any parameters.

Return value

Name	Type
countername	string

statistics.laser runtime.getunit

Unit of measurements

MODELS All

This method does not require any parameters.

Return value

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

statistics.laser.plate01.bank01.runtimeseconds.getname

Name of the counter

MODELS All

This method does not require any parameters.

Return value

Name	Type
countername	string

statistics.laser.plate01.bank01.runtimeseconds.getunit

Unit of measurements

MODELS All

This method does not require any parameters.

Return value

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

statistics.laser.plate01.bank02.runtimeseconds.getname

Name of the counter

MODELS All

This method does not require any parameters.

Return value

Name	Type
countername	string

statistics.laser.plate01.bank02.runtimeseconds.getunit

Unit of measurements

MODELS All

This method does not require any parameters.

Return value

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

statistics.laser.plate02.bank01.runtimeseconds.getname

Name of the counter

MODELS All

This method does not require any parameters.

Return value

Name	Type
countername	string

statistics.laser.plate02.bank01.runtimeseconds.getunit

Unit of measurements

MODELS All

This method does not require any parameters.

Return value

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

statistics.laser.plate02.bank02.runtimeseconds.getname

Name of the counter

MODELS All

This method does not require any parameters.

Return value

Name	Type
countername	string

statistics.laser.plate02.bank02.runtimeseconds.getunit

Unit of measurements

MODELS All

This method does not require any parameters.

Return value

Name	Type
unit	enum

Values
"none"
"hours"
"minutes"
"seconds"
"number"
"percent"

statistics.listcounters

List all counter names

MODELS

 All

This method does not require any parameters.

Return value

Name	Type																		
counterlist	[{ }]																		
	{ }																		
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>name</td><td>string</td></tr><tr><td>value</td><td>{ }</td></tr></table>	Name	Type	name	string	value	{ }												
Name	Type																		
name	string																		
value	{ }																		
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }												
Name	Type																		
base	int																		
hints	{ }																		
unit	enum																		
	<table><tr><th>Name</th><th>Type</th></tr><tr><td></td><td></td></tr><tr><th>Values</th><td></td></tr><tr><td>"none"</td><td></td></tr><tr><td>"hours"</td><td></td></tr><tr><td>"minutes"</td><td></td></tr><tr><td>"seconds"</td><td></td></tr><tr><td>"number"</td><td></td></tr><tr><td>"percent"</td><td></td></tr></table>	Name	Type			Values		"none"		"hours"		"minutes"		"seconds"		"number"		"percent"	
Name	Type																		
Values																			
"none"																			
"hours"																			
"minutes"																			
"seconds"																			
"number"																			
"percent"																			

statistics.maintenance.actuator.getname

Name of the counter

MODELS

 All

This method does not require any parameters.

Return value

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

countername

string

statistics.maintenance.actuator.getunit

Unit of measurements

MODELS

All

This method does not require any parameters.

Return value

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

statistics.maintenance.colorwheel.getname

Name of the counter

MODELS

All

This method does not require any parameters.

Return value

Name	Type
countername	string

statistics.maintenance.colorwheel.getunit

Unit of measurements

MODELS

All

This method does not require any parameters.

Return value

Name	Type
unit	enum
Values	
"none"	
"hours"	
"minutes"	

"seconds"
"number"
"percent"

statistics.maintenance.phosphorwheel.getname

Name of the counter

MODELS All

This method does not require any parameters.

Return value

Name	Type
countername	string

statistics.maintenance.phosphorwheel.getunit

Unit of measurements

MODELS All

This method does not require any parameters.

Return value

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

statistics.projector on.getname

Name of the counter

MODELS All

This method does not require any parameters.

Return value

Name	Type
countername	string

statistics.projector on.getunit

Unit of measurements

MODELS

All

This method does not require any parameters.

Return value

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

statistics.projector uptime.getname

Name of the counter

MODELS

All

This method does not require any parameters.

Return value

Name	Type
countername	string

statistics.projector uptime.getunit

Unit of measurements

MODELS

All

This method does not require any parameters.

Return value

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

statistics.projector.getname

Name of the counter

MODELS All

This method does not require any parameters.

Return value

Name	Type
countername	string

statistics.projector.getunit

Unit of measurements

MODELS All

This method does not require any parameters.

Return value

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

system.activity

Signal user activity (resets timeout countdown timers)

MODELS All

This method does not require any parameters.

This method has no return value.

system.boards.getboardlist

Retrieve list of detected boards

MODELS All

This method does not require any parameters.

Return value

Name	Type
boards	[string]

system.boards.getmissingboardlist

Retrieve list of missing boards

MODELS All

This method does not require any parameters.

Return value

Name	Type
boards	[string]

system.date.getlocaldate

Returns the system date as local time.

MODELS All

This method does not require any parameters.

Return value

Name	Type																																																												
date	{ }																																																												
<table><tr><th>Name</th><th>Type</th></tr><tr><td>year</td><td>{ }</td></tr></table>		Name	Type	year	{ }																																																								
Name	Type																																																												
year	{ }																																																												
<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr><tr><td>month</td><td>{ }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr><tr><td>day</td><td>{ }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr><tr><td>hour</td><td>{ }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr></table></td></tr></table></td></tr></table></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type		month	{ }	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr><tr><td>day</td><td>{ }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr><tr><td>hour</td><td>{ }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr></table></td></tr></table></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type		day	{ }	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr><tr><td>hour</td><td>{ }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr></table></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type		hour	{ }	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																																																						
Name	Type																																																												
base	int																																																												
hints	{ }																																																												
<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																											
Name	Type																																																												
month	{ }																																																												
<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr><tr><td>day</td><td>{ }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr><tr><td>hour</td><td>{ }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr></table></td></tr></table></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type		day	{ }	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr><tr><td>hour</td><td>{ }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr></table></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type		hour	{ }	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																	
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																																																						
Name	Type																																																												
base	int																																																												
hints	{ }																																																												
<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																											
Name	Type																																																												
day	{ }																																																												
<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr><tr><td>hour</td><td>{ }</td></tr><tr><td colspan="2"><table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr></table></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type		hour	{ }	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																	
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																																																						
Name	Type																																																												
base	int																																																												
hints	{ }																																																												
<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																											
Name	Type																																																												
hour	{ }																																																												
<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr></table>			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																	
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }																																																						
Name	Type																																																												
base	int																																																												
hints	{ }																																																												
<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																																																											
Name	Type																																																												

	<table><tr><th>Name</th><th>Type</th></tr><tr><td>minute</td><td>{ }</td></tr></table>	Name	Type	minute	{ }		
Name	Type						
minute	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>second</td><td>{ }</td></tr></table>	Name	Type	second	{ }		
Name	Type						
second	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

system.date.getsystemdate

Returns the system date as UTC time.

MODELS All

This method does not require any parameters.

Return value

	<table><tr><th>Name</th><th>Type</th></tr><tr><td>date</td><td>{ }</td></tr></table>	Name	Type	date	{ }		
Name	Type						
date	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>year</td><td>{ }</td></tr></table>	Name	Type	year	{ }		
Name	Type						
year	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>month</td><td>{ }</td></tr></table>	Name	Type	month	{ }		
Name	Type						
month	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>day</td><td>{ }</td></tr></table>	Name	Type	day	{ }		
Name	Type						
day	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>hour</td><td>{ }</td></tr></table>	Name	Type	hour	{ }		
Name	Type						
hour	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

minute { }

Name	Type
base	int
hints	{ }

second { }

Name	Type
base	int
hints	{ }

Name	Type
base	int
hints	{ }

system.getidentifications

Description not provided

MODELS All

This method does not require any parameters.

Return value

Name	Type						
identification	[{ }]						
	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>key</td><td>string</td></tr><tr><td>value</td><td>string</td></tr></table>	Name	Type	key	string	value	string
Name	Type						
key	string						
value	string						

system.getsystemdate

Description not provided

MODELS All

This method does not require any parameters.

Return value

Name	Type												
date	{ }												
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>year</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr></table>	Name	Type	year	{ }		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type												
year	{ }												
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }						
Name	Type												
base	int												
hints	{ }												
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>month</td><td>{ }</td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr></table>	Name	Type	month	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type												
month	{ }												
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type										
Name	Type												

		base	int
		hints	{ }
day	{ }		
		Name	Type
		Name	Type
		base	int
		hints	{ }
		Name	Type
hour	{ }		
		Name	Type
		base	int
		hints	{ }
		Name	Type
minute	{ }		
		Name	Type
		base	int
		hints	{ }
		Name	Type
second	{ }		
		Name	Type
		base	int
		hints	{ }
		Name	Type

system.gotoeco

Set the device in the eco state

MODELS All

This method does not require any parameters.

This method has no return value.

system.gotoready

Set the device from standby state to ready state

MODELS All

This method does not require any parameters.

This method has no return value.

system.license.file.saveregistrationtousb

Save registration file to usb memory device

MODELS All

This method does not require any parameters.

This method has no return value.

system.license.file.uploadfromusb

Upload license from usb memory device

MODELS All

This method does not require any parameters.

This method has no return value.

system.listresetdomains

Returns the list of available reset domains

MODELS All

This method does not require any parameters.

Return value

Name	Type
domains	[enum]
	Values
	"Dmx"
	"Environment"
	"Gpio"
	"Gsm"
	"Illumination"
	"ImageBlend"
	"ImageCapture"
	"ImageConnector"
	"ImageConvergence"
	"ImageDisplay"
	"ImageFeatures"
	"ImageOrientation"
	"ImageRealColor"
	"ImageResolution"
	"ImageSource"
	"ImageStereo"
	"ImageTestPattern"
	"ImageTiltedBrightness"
	"ImageUniformity"
	"ImageWarp"
	"LightMeasurement"

"LightSteering"
"LinkedServices"
"Macro"
"MediaPlayer"
"Network"
"Optics"
"Peripheral"
"PJLink"
"Profiles"
"Scheduler"
"Screen"
"Sntp"
"System"
"Time"
"User"
"UserInterface"

system.poweroff

Power off the unit

MODELS All

This method does not require any parameters.

This method has no return value.

system.poweron

Power on the unit

MODELS All

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.

MODELS All

Parameters

Name	Type
domains	[enum]
Values	
"Dmx"	

```
"Environment"  
"Gpio"  
"Gsm"  
"Illumination"  
"ImageBlend"  
"ImageCapture"  
"ImageConnector"  
"ImageConvergence"  
"ImageDisplay"  
"ImageFeatures"  
"ImageOrientation"  
"ImageRealColor"  
"ImageResolution"  
"ImageSource"  
"ImageStereo"  
"ImageTestPattern"  
"ImageTiltedBrightness"  
"ImageUniformity"  
"ImageWarp"  
"LightMeasurement"  
"LightSteering"  
"LinkedServices"  
"Macro"  
"MediaPlayer"  
"Network"  
"Optics"  
"Peripheral"  
"PJLink"  
"Profiles"  
"Scheduler"  
"Screen"  
"Snmp"  
"System"  
"Time"  
"User"  
"UserInterface"
```

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. Execute this function with caution, as it might remove the projector's hostname, requiring physical access to the projector.

MODELS

All

This method does not require any parameters.

Return value

Name	Type
domains	[enum]
	Values
	"Dmx"
	"Environment"
	"Gpio"
	"Gsm"
	"Illumination"
	"ImageBlend"
	"ImageCapture"
	"ImageConnector"
	"ImageConvergence"
	"ImageDisplay"
	"ImageFeatures"
	"ImageOrientation"
	"ImageRealColor"
	"ImageResolution"
	"ImageSource"
	"ImageStereo"
	"ImageTestPattern"
	"ImageTiltedBrightness"
	"ImageUniformity"
	"ImageWarp"
	"LightMeasurement"
	"LightSteering"
	"LinkedServices"
	"Macro"
	"MediaPlayer"
	"Network"
	"Optics"
	"Peripheral"
	"PJLink"
	"Profiles"
	"Scheduler"
	"Screen"
	"Snmp"
	"System"
	"Time"

"User"

"UserInterface"

ui.leds.setstate

Set the state and color of the key LEDs

MODELS

All

Parameters

Name	Type
key	enum
Values	
"Back"	
"Down"	
"Input"	
"Left"	
"Lens"	
"Menu"	
"Ok"	
"OsdOff"	
"Pattern"	
"Right"	
"Up"	
state	enum
Values	
"Off"	
"White"	
"Blue"	

This method has no return value.

ui.settings.get

Get the value of the specified key

MODELS

All

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.

MODELS

All

Parameters

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

Return value

Name	Type						
dictionary	[{ }]						
{ }							
<table><tr><th>Name</th><th>Type</th></tr><tr><td>key</td><td>string</td></tr><tr><td>value</td><td>string</td></tr></table>		Name	Type	key	string	value	string
Name	Type						
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.

MODELS

All

Parameters

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

Return value

Name	Type						
dictionary	[{ }]						
{ }							
<table><tr><th>Name</th><th>Type</th></tr><tr><td>key</td><td>string</td></tr><tr><td>value</td><td>string</td></tr></table>		Name	Type	key	string	value	string
Name	Type						
key	string						
value	string						

ui.settings.keys

Return a list of all the keys

MODELS All

This method does not require any parameters.

Return value

Name	Type
keys	[string]

ui.settings.list

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

MODELS All

This method does not require any parameters.

Return value

Name	Type
dict	[{ }]
	{ }

Name	Type
key	string
value	string

ui.settings.remove

Remove the specified key and value

MODELS All

Parameters

Name	Type
key	string

This method has no return value.

ui.settings.set

Set the key to the specified value

MODELS All

Parameters

Name	Type
key	string
value	string

This method has no return value.

ui.togglelensmenu

Toggle lens menu on or off

MODELS All

This method does not require any parameters.

This method has no return value.

ui.togglemenu

Toggle menu on/off

MODELS All

This method does not require any parameters.

This method has no return value.

ui.toggleosd

Toggle on screen display on or off

MODELS All

This method does not require any parameters.

This method has no return value.

ui.togglepatternmenu

Toggle the pattern menu on or off

MODELS All

This method does not require any parameters.

This method has no return value.

ui.togglesourcemenu

Toggle the source menu on or off

MODELS All

This method does not require any parameters.

This method has no return value.

ui.togglestealthmode

[DEPRECATED]

MODELS All

This method does not require any parameters.

This method has no return value.

ui.touchscreen.tapped

Called from touchpanel when the touchpanel is in Test mode

MODELS All

Parameters

Name	Type						
x	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
y	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

This method has no return value.

user.admin.changegroup

Change the user group

MODELS All

Parameters

Name	Type
newgroup	string

This method has no return value.

user.admin.changepassword

Change the password

MODELS All

Parameters

Name	Type
newpassword	string

This method has no return value.

user.admin.changepincode

Change the 6-digit PIN code

MODELS All

Parameters

Name	Type
newpincode	{ }
Name	Type
base	int
hints	{ }
Name	Type

This method has no return value.

user.admin.changepublickey

Change the public key

MODELS All

Parameters

Name	Type
newpublickey	string

This method has no return value.

user.admin.changeusername

Change the username

MODELS All

Parameters

Name	Type
newusername	string

Return value

Name	Type
newuserslug	string

user.admin.disable

Disable the user

MODELS All

This method does not require any parameters.

This method has no return value.

user.admin.enable

Enable the user

MODELS All

This method does not require any parameters.

This method has no return value.

user.admin.invalidatetokens

Invalidate all tokens

MODELS All

This method does not require any parameters.

This method has no return value.

user.admin.logouteverywhere

Invalidate all tokens and log out of all active sessions

MODELS All

This method does not require any parameters.

This method has no return value.

user.admin.removepassword

Remove the password

MODELS All

This method does not require any parameters.

This method has no return value.

user.admin.removepincode

Remove the 6-digit PIN code

MODELS All

This method does not require any parameters.

This method has no return value.

user.currentuser.changegroup

Change the user group

MODELS

 All

Parameters

Name	Type
newgroup	string

This method has no return value.

user.currentuser.changepassword

Change the password

MODELS

 All

Parameters

Name	Type
newpassword	string

This method has no return value.

user.currentuser.changepincode

Change the 6-digit PIN code

MODELS

 All

Parameters

Name	Type						
newpincode	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

This method has no return value.

user.currentuser.changepublickey

Change the public key

MODELS

 All

Parameters

Name	Type
newpublickey	string

This method has no return value.

user.currentuser.changeusername

Change the username

MODELS All

Parameters

Name	Type
newusername	string

Return value

Name	Type
newuserslug	string

user.currentuser.disable

Disable the user

MODELS All

This method does not require any parameters.

This method has no return value.

user.currentuser.enable

Enable the user

MODELS All

This method does not require any parameters.

This method has no return value.

user.currentuser.invalidatetokens

Invalidate all tokens

MODELS All

This method does not require any parameters.

This method has no return value.

user.currentuser.logouteverywhere

Invalidate all tokens and log out of all active sessions

MODELS All

This method does not require any parameters.

This method has no return value.

user.currentuser.removepassword

Remove the password

MODELS All

This method does not require any parameters.

This method has no return value.

user.currentuser.removepincode

Remove the 6-digit PIN code

MODELS All

This method does not require any parameters.

This method has no return value.

user.poweruser.changegroup

Change the user group

MODELS All

Parameters

Name	Type
newgroup	string

This method has no return value.

user.poweruser.changepassword

Change the password

MODELS All

Parameters

Name	Type
newpassword	string

This method has no return value.

user.poweruser.changepincode

Change the 6-digit PIN code

MODELS All

Parameters

Name	Type
newpincode	{ }

Name	Type
base	int
hints	{ }

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

This method has no return value.

user.poweruser.changepublickey

Change the public key

MODELS All

Parameters

Name	Type
newpublickey	string

This method has no return value.

user.poweruser.changeusername

Change the username

MODELS All

Parameters

Name	Type
newusername	string

Return value

Name	Type
newuserslug	string

user.poweruser.disable

Disable the user

MODELS All

This method does not require any parameters.

This method has no return value.

user.poweruser.enable

Enable the user

MODELS All

This method does not require any parameters.

This method has no return value.

user.poweruser.invalidatetokens

Invalidate all tokens

MODELS All

This method does not require any parameters.

This method has no return value.

user.poweruser.logouteverywhere

Invalidate all tokens and log out of all active sessions

MODELS All

This method does not require any parameters.

This method has no return value.

user.poweruser.removepassword

Remove the password

MODELS All

This method does not require any parameters.

This method has no return value.

user.poweruser.removepincode

Remove the 6-digit PIN code

MODELS All

This method does not require any parameters.

This method has no return value.

user.resetadministrator

Reset the administrator account using the response provided by barco

MODELS All

Parameters

Name	Type
response	{ }
Name	Type
base	int

hints { }

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

This method has no return value.

user.testuser.changegroup

Change the user group

MODELS All

Parameters

Name	Type
newgroup	string

This method has no return value.

user.testuser.changepassword

Change the password

MODELS All

Parameters

Name	Type
newpassword	string

This method has no return value.

user.testuser.changepincode

Change the 6-digit PIN code

MODELS All

Parameters

Name	Type
newpincode	{ }
Name	Type
base	int
hints	{ }
Name	Type

This method has no return value.

user.testuser.changepublickey

Change the public key

MODELS All

Parameters

Name	Type
newpublickey	string

This method has no return value.

user.testuser.changeusername

Change the username

MODELS All

Parameters

Name	Type
newusername	string

Return value

Name	Type
newuserslug	string

user.testuser.disable

Disable the user

MODELS All

This method does not require any parameters.

This method has no return value.

user.testuser.enable

Enable the user

MODELS All

This method does not require any parameters.

This method has no return value.

user.testuser.invalidatetokens

Invalidate all tokens

MODELS All

This method does not require any parameters.

This method has no return value.

user.testuser.logouteverywhere

Invalidate all tokens and log out of all active sessions

MODELS All

This method does not require any parameters.

This method has no return value.

user.testuser.removepassword

Remove the password

MODELS All

This method does not require any parameters.

This method has no return value.

user.testuser.removepincode

Remove the 6-digit PIN code

MODELS All

This method does not require any parameters.

This method has no return value.

user.user.changegroup

Change the user group

MODELS All

Parameters

Name	Type
newgroup	string

This method has no return value.

user.user.changepassword

Change the password

MODELS All

Parameters

Name	Type
newpassword	string

This method has no return value.

user.user.changepincode

Change the 6-digit PIN code

MODELS

All

Parameters

Name	Type						
newpincode	{ }						
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>int</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	int	hints	{ }
Name	Type						
base	int						
hints	{ }						
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type				
Name	Type						

This method has no return value.

user.user.changepublickey

Change the public key

MODELS

All

Parameters

Name	Type
newpublickey	string

This method has no return value.

user.user.changeusername

Change the username

MODELS

All

Parameters

Name	Type
newusername	string

Return value

Name	Type
newuserslug	string

user.user.disable

Disable the user

MODELS

All

This method does not require any parameters.

This method has no return value.

user.user.enable

Enable the user

MODELS All

This method does not require any parameters.

This method has no return value.

user.user.invalidatetokens

Invalidate all tokens

MODELS All

This method does not require any parameters.

This method has no return value.

user.user.logouteverywhere

Invalidate all tokens and log out of all active sessions

MODELS All

This method does not require any parameters.

This method has no return value.

user.user.removepassword

Remove the password

MODELS All

This method does not require any parameters.

This method has no return value.

user.user.removepincode

Remove the 6-digit PIN code

MODELS All

This method does not require any parameters.

This method has no return value.

Signals

Alphabetical list of all signals

loggedout

Will be raised whenever the user is logged out

MODELS All		
Name	Type	Description
reason	enum	
Values		
"Disable"		
"InternalError"		
"LogoutEverywhere"		
"Reauthentication"		
"Reboot"		
"Remove"		
"Request"		
"Reset"		
"Restore"		

modelupdated

Signals whenever functionality for an object appears or disappears

MODELS All		
Name	Type	Description
object	string	The object name (JSON-RPC dot-notation)
newobject	bool	True if functionality is added and false if functionality is removed
reason	string	The reason for this ModelUpdated signal, either an object was added, removed or changed.
visibilityaccesslevel	enum	Minimum access level for this object to be visible. It will include the lowest access level of all the children of a node because that determines if a node is visible for a user level.
Values		
"UNAUTHENTICATED_END_USER"		
"END_USER"		
"POWER_USER"		
"ADMINISTRATOR"		
"SERVICE_PARTNER"		
"MANUFACTURING"		
"DEVELOPMENT"		
"INACCESSIBLE"		
accesslevel	enum	Minimum access level for this object
Values		

"UNAUTHENTICATED_END_USER"

"END_USER"

"POWER_USER"

"ADMINISTRATOR"

"SERVICE_PARTNER"

"MANUFACTURING"

"DEVELOPMENT"

"INACCESSIBLE"

dmx.shutdownrequest

Description not provided

MODELS All

This signal does contain any arguments.

image.brilliantcolor.listchanged

Will be raised when BrilliantColor environment has changed

MODELS All

This signal does contain any arguments.

image.connector.displayport.edid.listchanged

Will be raised when list of EDIDs changed

MODELS All

This signal does contain any arguments.

image.connector.hdmi.edid.listchanged

Will be raised when list of EDIDs changed

MODELS All

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

MODELS

All

This signal does contain any arguments.

image.processing.warpgrid.gridchanged

Description not provided

MODELS

All

Name	Type	Description																												
grid	[{ }]																													
	{ }																													
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>x</td><td>{ }</td></tr></table>	Name	Type	x	{ }																									
Name	Type																													
x	{ }																													
	<table><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td></tr><tr><td>y</td><td>{ }</td><td></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table></td><td></td></tr><tr><td></td><td><table><tr><th>Name</th><th>Type</th></tr></table></td><td></td></tr></table>		<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }		<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	y	{ }			<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }			<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type	
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																							
Name	Type																													
base	float																													
hints	{ }																													
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																											
Name	Type																													
y	{ }																													
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>base</td><td>float</td></tr><tr><td>hints</td><td>{ }</td></tr></table>	Name	Type	base	float	hints	{ }																							
Name	Type																													
base	float																													
hints	{ }																													
	<table><tr><th>Name</th><th>Type</th></tr></table>	Name	Type																											
Name	Type																													

image.testpattern.added

Description not provided

MODELS

All

Name	Type	Description								
pattern	{ }	The ID of the new pattern that was added								
	<table><tr><th>Name</th><th>Type</th></tr><tr><td>name</td><td>string</td></tr><tr><td>location</td><td>string</td></tr><tr><td>id</td><td>string</td></tr></table>	Name	Type	name	string	location	string	id	string	
Name	Type									
name	string									
location	string									
id	string									

```

above      bool
internal   bool
properties [ { } ]

```

Name	Type
key	string
value	string

image.testpattern.changed

Description not provided

MODELS All

Name	Type	Description
id	string	The unique pattern id of the pattern that have changed
properties	[{ }]	{ }

Name	Type
key	string
value	string

image.testpattern.file.listchanged

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

MODELS All

This signal does contain any arguments.

image.testpattern.removed

Description not provided

MODELS

All

Name	Type	Description
pattern	{ }	The ID of the pattern that was removed

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

{ }	
key	string
value	string

keydispatcher.keyevent

Raised when the user has pressed a physical key on an input device.

MODELS All

Name	Type	Description
key	string	The logical key name that was pressed
type	enum	Type of event: press, repeat, release, clicked. Values "PRESS" "RELEASE" "REPEAT" "CLICKED"
source	string	The source of the event, e.g remote control or keypad

network.added

Raised when a new device has been added

MODELS All

Name	Type	Description
id	string	The logical device id of the new device, e.g: 'wifi1'

network.removed

Raised when a device has been removed

MODELS All

Name	Type	Description
id	string	The logical device id of the device that was removed, e.g: 'wired2'

notification.caution.dismissed

Signal the dismissal of a notification of the given severity level

MODELS All

This signal does contain any arguments.		
---	--	--

notification.caution.emitted

Signal the emission of a new notification of the given severity level

MODELS All

This signal does contain any arguments.		
---	--	--

notification.dismissed

Description not provided

MODELS All

--	--	--

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

notification.emitted

Description not provided

MODELS

All

Name	Type	Description
notification	{ }	The new notification
NameType		
severity	enum	
Values		
"INFO "		
"CAUTION "		
"WARNING "		
"ERROR "		
id	string	
code	string	
timestamp	string	
arguments	[string]	
message	string	
timeout	string	
visibility	string	
doc	string	
actions	[enum]	
Values		
"NONE "		
"OK "		
"CANCEL "		
"YES "		
"NO "		
"SUPPRESS "		
messageTemplate	string	

notification.error.dismissed

Signal the dismissal of a notification of the given severity level

MODELS All

This signal does contain any arguments.

notification.error.emitted

Signal the emission of a new notification of the given severity level

MODELS All

This signal does contain any arguments.

notification.info.dismissed

Signal the dismissal of a notification of the given severity level

MODELS All

This signal does contain any arguments.

notification.info.emitted

Signal the emission of a new notification of the given severity level

MODELS All

This signal does contain any arguments.

notification.warning.dismissed

Signal the dismissal of a notification of the given severity level

MODELS All

This signal does contain any arguments.

notification.warning.emitted

Signal the emission of a new notification of the given severity level

MODELS All

This signal does contain any arguments.

profile.profileactivated

Fired when a profile is activated

MODELS All

Name	Type	Description
name	string	The name of the activated profile

system.date.systemdatechanged

Signals if date/time has changed.

MODELS

All

This signal does contain any arguments.

system.identificationchanged

Will be raised whenever an identification is changed

MODELS

All

Name	Type	Description
identification	string	

system.license.licensechanged

Description not provided

MODELS

All

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'.

MODELS

All

Name	Type	Description
domains	[enum]	
		Values
		"Dmx"
		"Environment"
		"Gpio"
		"Gsm"
		"Illumination"
		"ImageBlend"
		"ImageCapture"
		"ImageConnector"
		"ImageConvergence"
		"ImageDisplay"
		"ImageFeatures"
		"ImageOrientation"
		"ImageRealColor"
		"ImageResolution"
		"ImageSource"
		"ImageStereo"
		"ImageTestPattern"

"ImageTiltedBrightness"

"ImageUniformity"

"ImageWarp"

"LightMeasurement"

"LightSteering"

"LinkedServices"

"Macro"

"MediaPlayer"

"Network"

"Optics"

"Peripheral"

"PJLink"

"Profiles"

"Scheduler"

"Screen"

"Sntp"

"System"

"Time"

"User"

"UserInterface"

ui.settings.added

Fired when a new key/value pair was added

MODELS All

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

ui.settings.changed

Fired when a key has an updated value

MODELS All

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

ui.settings.removed

Fired when a key was removed

MODELS All

Name	Type	Description
key	string	The key name

Files

Alphabetical list of all file end points.

bio.logger.transfer

Download notification log files

MODELS All	
Upload	Download
No	Yes

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

 

```
pulse:~$ curl -O -J http://192.168.1.100/api/bio/logger/transfer
```

Example of file download using the **PowerShell** on **Windows**.



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

image.connector.edid.transfer

Upload and download EDID files

MODELS All	
Upload	Download
Yes	Yes

Example of file upload using the `curl` program.

 

```
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 `-O` and `-J` option makes `curl` use the file name suggestion of the server and will overwrite a file if it already exists.

 

```
pulse:~$ curl -O -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

MODELS

All

Upload

Download

Yes

Yes

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 `-O` and `-J` option makes `curl` use the file name suggestion of the server and will overwrite a file if it already exists.

 

```
pulse:~$ curl -O -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

MODELS

All

Upload

Download

Yes

Yes

Example of file upload using the `curl` program.

 

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

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

 

```
pulse:~$ curl -O -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

MODELS All	
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 `-O` and `-J` option makes `curl` use the file name suggestion of the server and will overwrite a file if it already exists.

 

```
pulse:~$ curl -O -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

MODELS All	
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 `-O` and `-J` option makes `curl` use the file name suggestion of the server and will overwrite a file if it already exists.

 

```
pulse:~$ curl -O -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

MODELS

All

Upload

Download

No

Yes

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

 

```
pulse:~$ curl -O -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
```

system.license.file.registration

Download registration file

MODELS

All

Upload

Download

No

Yes

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

 

```
pulse:~$ curl -O -J http://192.168.1.100/api/system/license/file/registration
```

Example of file download using the **PowerShell** on **Windows**.



```
PS C:\Users\pulse> Invoke-WebRequest -Uri http://192.168.1.100/api/system/license/file/registration -
Method Get -OutFile registration.dat
```

ui.capture.lcd

Download screen capture of LCD

MODELS

All

Upload

Download

No

Yes

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



```
pulse:~$ curl -O -J http://192.168.1.100/api/ui/capture/lcd
```

Example of file download using the **PowerShell** on **Windows**.



```
PS C:\Users\pulse> Invoke-WebRequest -Uri http://192.168.1.100/api/ui/capture/lcd -Method Get -OutFile  
lcd.dat
```

ui.capture.osd

Download screen capture of OSD

MODELS		All
Upload	Download	
No	Yes	

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



```
pulse:~$ curl -O -J http://192.168.1.100/api/ui/capture/osd
```

Example of file download using the **PowerShell** on **Windows**.



```
PS C:\Users\pulse> Invoke-WebRequest -Uri http://192.168.1.100/api/ui/capture/osd -Method Get -OutFile  
osd.dat
```

ui.splashscreen.file

Upload splash screen for LCD - 800x480 PNG,JPG

MODELS		All
Upload	Download	
Yes	No	

Example of file upload using the **curl** program.



```
pulse:~$ curl -F file=@splashscreen.dat http://192.168.1.100/api/ui/splashscreen/file
```

Property index

backup.auto.error	44	backup.auto.ipaddress	44
backup.auto.outofsyncdate	44	backup.auto.state	44
backup.auto.syncstate	45	diagnostics.selftest.progress	45
diagnostics.selftest.result	45	diagnostics.selftest.status	46
dmx.artnet	47	dmx.artnetnet	47
dmx.artnetuniverse	47	dmx.mode	47
dmx.monitor.channel01.function	48	dmx.monitor.channel01.offset	48
dmx.monitor.channel01.value	48	dmx.monitor.channel02.function	48
dmx.monitor.channel02.offset	49	dmx.monitor.channel02.value	49
dmx.monitor.connectionstate.active	49	dmx.shutdown	49
dmx.shutdowntimeout	50	dmx.startchannel	50
environment.alarmstate	50	firmware.availablepackages	50
firmware.component.backplane-boot-redstar.actualversion	51	firmware.component.backplane-boot-redstar.displayname	51
firmware.component.backplane-boot-redstar.error	51	firmware.component.backplane-boot-redstar.mismatch	51
firmware.component.backplane-boot-redstar.progress	52	firmware.component.backplane-boot-redstar.requiredversion	52
firmware.component.backplane-boot-redstar.status	53	firmware.component.backplane-redstar-fpga.actualversion	53
firmware.component.backplane-redstar-fpga.displayname	53	firmware.component.backplane-redstar-fpga.error	54
firmware.component.backplane-redstar-fpga.mismatch	54	firmware.component.backplane-redstar-fpga.progress	54
firmware.component.backplane-redstar-fpga.requiredversion	54	firmware.component.backplane-redstar-fpga.status	54
firmware.component.backplane-run-redstar.actualversion	55	firmware.component.backplane-run-redstar.displayname	55
firmware.component.backplane-run-redstar.error	56	firmware.component.backplane-run-redstar.mismatch	56
firmware.component.backplane-run-redstar.progress	56	firmware.component.backplane-run-redstar.requiredversion	56
firmware.component.backplane-run-redstar.status	56	firmware.component.components	57
firmware.component.formatter-1 dlp-fpga.actualversion	57	firmware.component.formatter-1 dlp-fpga.displayname	57
firmware.component.formatter-1 dlp-fpga.error	57	firmware.component.formatter-1 dlp-fpga.mismatch	58
firmware.component.formatter-1 dlp-fpga.progress	58	firmware.component.formatter-1 dlp-fpga.requiredversion	58
firmware.component.formatter-1 dlp-fpga.status	58	firmware.component.formatter-ddp442x-sequences.actualversion	59
firmware.component.formatter-ddp442x-sequences.displayname	59	firmware.component.formatter-ddp442x-sequences.error	59
firmware.component.formatter-ddp442x-sequences.mismatch	59	firmware.component.formatter-ddp442x-sequences.progress	60
firmware.component.formatter-ddp442x-sequences.requiredversion	60	firmware.component.formatter-ddp442x-sequences.status	61
firmware.component.formatter-rallar-boot.actualversion	61	firmware.component.formatter-rallar-boot.displayname	61
firmware.component.formatter-rallar-boot.error	61	firmware.component.formatter-rallar-boot.mismatch	62
firmware.component.formatter-rallar-boot.progress	62	firmware.component.formatter-rallar-boot.requiredversion	62
firmware.component.formatter-rallar-boot.status	62	firmware.component.formatter-rallar-run.actualversion	63
firmware.component.formatter-rallar-run.displayname	63		63
	64		64

Pulse API		Reference Guide	Version 2.5
firmware.component.formatter-rallar-run.mismatch	64	firmware.component.formatter-rallar-run.error	64
firmware.component.formatter-rallar-run.requiredversion	64	firmware.component.formatter-rallar-run.progress	65
	65	firmware.component.formatter-rallar-run.status	65
firmware.component.green-controller-boot-v1.actualversion	65	firmware.component.green-controller-boot-v1.displayname	65
firmware.component.green-controller-boot-v1.error	66		66
firmware.component.green-controller-boot-v1.progress	66	firmware.component.green-controller-boot-v1.mismatch	66
	67	firmware.component.green-controller-boot-v1.requiredversion	67
firmware.component.green-controller-boot-v1.status	67		67
firmware.component.green-controller-run-v1.displayname	67	firmware.component.green-controller-run-v1.actualversion	68
firmware.component.green-controller-run-v1.mismatch	68	firmware.component.green-controller-run-v1.error	68
	68	firmware.component.green-controller-run-v1.progress	69
firmware.component.green-controller-run-v1.requiredversion	69	firmware.component.green-controller-run-v1.status	69
firmware.component.ld1-megmeet-psu.actualversion	69	firmware.component.ld1-megmeet-psu.displayname	69
	70	firmware.component.ld1-megmeet-psu.mismatch	70
firmware.component.ld1-megmeet-psu.error	70		70
firmware.component.ld1-megmeet-psu.progress	71	firmware.component.ld1-megmeet-psu.requiredversion	71
firmware.component.ld1-megmeet-psu.status	71	firmware.component.ld2-megmeet-psu.actualversion	71
firmware.component.ld2-megmeet-psu.displayname	71	firmware.component.ld2-megmeet-psu.error	72
firmware.component.ld2-megmeet-psu.mismatch	72	firmware.component.ld2-megmeet-psu.progress	72
firmware.component.ld2-megmeet-psu.requiredversion	72	firmware.component.ld2-megmeet-psu.status	73
	73	firmware.component.linux.displayname	73
firmware.component.linux.actualversion	73	firmware.component.linux.mismatch	73
firmware.component.linux.error	74	firmware.component.linux.requiredversion	74
firmware.component.linux.progress	74		74
firmware.component.linux.status	74	firmware.component.mysticum-image-processing-3-mk22-boot.actualversion	74
firmware.component.mysticum-image-processing-3-mk22-boot.displayname	74	firmware.component.mysticum-image-processing-3-mk22-boot.error	75
	75		75
firmware.component.mysticum-image-processing-3-mk22-boot.mismatch	75	firmware.component.mysticum-image-processing-3-mk22-boot.progress	76
	76		76
firmware.component.mysticum-image-processing-3-mk22-boot.requiredversion	76	firmware.component.mysticum-image-processing-3-mk22-boot.status	76
firmware.component.mysticum-image-processing-3-mk22-fpga.actualversion	77	firmware.component.mysticum-image-processing-3-mk22-fpga.displayname	77
	77		77
firmware.component.mysticum-image-processing-3-mk22-fpga.error	78	firmware.component.mysticum-image-processing-3-mk22-fpga.mismatch	78
firmware.component.mysticum-image-processing-3-mk22-fpga.progress	78	firmware.component.mysticum-image-processing-3-mk22-fpga.requiredversion	78
	78		79
firmware.component.mysticum-image-processing-3-mk22-fpga.status	79	firmware.component.mysticum-image-processing-3-mk22-run.actualversion	79
firmware.component.mysticum-image-processing-3-mk22-run.displayname	79	firmware.component.mysticum-image-processing-3-mk22-run.error	80
	80		80
firmware.component.mysticum-image-processing-3-mk22-run.mismatch	80	firmware.component.mysticum-image-processing-3-mk22-run.progress	81
firmware.component.mysticum-image-processing-3-mk22-run.requiredversion	81	firmware.component.mysticum-image-processing-3-mk22-run.status	81
	81		81
firmware.component.redstar-fan-motor-control-board-fpga.actualversion	82	firmware.component.redstar-fan-motor-control-board-fpga.displayname	82
firmware.component.redstar-fan-motor-control-board-fpga.error	82	firmware.component.redstar-fan-motor-control-board-fpga.mismatch	82
	82		83
firmware.component.redstar-fan-motor-control-	83	firmware.component.redstar-fan-motor-control-board-fpga.requiredversion	83

Pulse API		Reference Guide	Version 2.5
board-fpga.progress	83	firmware.component.redstar-fan-motor-control-boot.actualversion	84
firmware.component.redstar-fan-motor-control-board-fpga.status	84	firmware.component.redstar-fan-motor-control-boot.error	84
firmware.component.redstar-fan-motor-control-boot.displayname	84	firmware.component.redstar-fan-motor-control-boot.progress	85
firmware.component.redstar-fan-motor-control-boot.mismatch	85	firmware.component.redstar-fan-motor-control-boot.status	85
firmware.component.redstar-fan-motor-control-boot.requiredversion	86	firmware.component.redstar-fan-motor-control-run.displayname	86
firmware.component.redstar-fan-motor-control-run.actualversion	86	firmware.component.redstar-fan-motor-control-run.mismatch	86
firmware.component.redstar-fan-motor-control-run.error	87	firmware.component.redstar-fan-motor-control-run.requiredversion	87
firmware.component.redstar-fan-motor-control-run.progress	87	firmware.component.status	88
firmware.component.redstar-fan-motor-control-run.status	88	firmware.component.u-boot.displayname	88
firmware.component.u-boot.actualversion	89	firmware.component.u-boot.mismatch	89
firmware.component.u-boot.error	90	firmware.component.u-boot.requiredversion	90
firmware.component.u-boot.progress	90	firmware.component.zynq-fpga.actualversion	91
firmware.component.u-boot.status	91	firmware.component.zynq-fpga.error	91
firmware.component.zynq-fpga.displayname	91	firmware.component.zynq-fpga.progress	92
firmware.component.zynq-fpga.mismatch	92	firmware.component.zynq-fpga.status	93
firmware.component.zynq-fpga.requiredversion	93	firmware.eula.requiretermagreement	93
firmware.error	93	firmware.firmwareversion	94
firmware.eula.termagreement	94	firmware.status	94
firmware.progress	94	gpio.gpo.trigger1.mode	95
gpio.gpo.trigger1.description	95	gpio.gpo.trigger1.reserved	95
gpio.gpo.trigger1.name	96	gpio.gpo.trigger2.description	96
gpio.gpo.trigger1.value	96	gpio.gpo.trigger2.name	97
gpio.gpo.trigger2.mode	97	gpio.gpo.trigger2.value	97
gpio.gpo.trigger2.reserved	97	gsm.iccid	97
gsm.data.status	98	gsm.signalstrength	98
gsm.imei	98	illumination.clo.availability	99
illumination.clo.allowed	99	illumination.clo.scale	99
illumination.clo.enable	100	illumination.clo.state	100
illumination.clo.setpoint	100	illumination.nosignal.dimming.power	101
illumination.nosignal.dimming.enable	101	illumination.sources.laser.controlledbyclo	102
illumination.sources.laser.actualpower	102	illumination.sources.laser.maxpower	102
illumination.sources.laser.ispowerlimited	103	illumination.sources.laser.power	103
illumination.sources.laser.minpower	103	illumination.sources.laser.powerprecision	103
illumination.sources.laser.powerlimitreason	104	illumination.sources.laser.timcuring.dmd.cureavailable	104
illumination.sources.laser.status	104	illumination.state	105
illumination.sources.laser.timcuring.dmd.state	105	image.brilliantcolor.mode	105
image.brightness	106	image.color.p7.custom.bluelum	106
image.color.p7.custom.bluegain	106	image.color.p7.custom.bluey	107
image.color.p7.custom.blux	107	image.color.p7.custom.cmywritable	107
	108	image.color.p7.custom.cyanlum	108
	109		109

Pulse API		Reference Guide	Version 2.5
image.color.p7.custom.cmyreadable	111	image.color.p7.custom.cyany	111
image.color.p7.custom.cyangain	113	image.color.p7.custom.greengain	113
image.color.p7.custom.cyanx	113	image.color.p7.custom.greenx	113
image.color.p7.custom.gainsavailable	114	image.color.p7.custom.luminancesavailable	114
image.color.p7.custom.greenlum	114	image.color.p7.custom.magentalum	118
image.color.p7.custom.greeny	118	image.color.p7.custom.magentay	119
image.color.p7.custom.magentagain	119	image.color.p7.custom.modes	119
image.color.p7.custom.magentax	120	image.color.p7.custom.redlum	120
image.color.p7.custom.mode	120	image.color.p7.custom.redy	124
image.color.p7.custom.redgain	124	image.color.p7.custom.rgbwritable	124
image.color.p7.custom.redx	125	image.color.p7.custom.whitegain	125
image.color.p7.custom.rgbcmygainsavailable	125	image.color.p7.custom.whitelum	126
image.color.p7.custom.target	126	image.color.p7.custom.whitetemperature	126
image.color.p7.custom.whitegainavailable	127	image.color.p7.custom.whitewritable	127
image.color.p7.custom.whitemode	127	image.color.p7.custom.whitey	127
image.color.p7.custom.whitetemperatureavailable	128	image.color.p7.custom.yellowlum	128
image.color.p7.custom.whitex	128	image.color.p7.custom.yellowy	128
image.color.p7.custom.yellowgain	129	image.color.p7.native.c1available	129
image.color.p7.custom.yellowx	129	image.color.p7.native.c2available	130
image.color.p7.native.c1	130	image.color.p7.native.normal.c1	130
image.color.p7.native.c2	130	image.color.p7.native.normal.c2	131
image.color.p7.native.list	131	image.color.p7.native.normal.rgbw	131
image.color.p7.native.normal.c1available	132	image.color.p7.native.rgbw	132
image.color.p7.native.normal.c2available	132	image.color.rgbmode.rgbmode	133
image.color.p7.native.normal.whiteavailable	133	image.connector.displayport.colorprimaries	133
image.color.p7.native.whiteavailable	134	image.connector.displayport.colorspace	134
image.connector.displayport.capabilities	134	image.connector.displayport.edid.selected	134
image.connector.displayport.colorprimariesavailable	135	image.connector.displayport.signalrange	135
image.connector.displayport.detectedsignal	135	image.connector.hdmi.colorprimaries	135
image.connector.displayport.name	136	image.connector.hdmi.colorspace	136
image.connector.hdmi.capabilities	137	image.connector.hdmi.edid.selected	137
image.connector.hdmi.colorprimariesavailable	138	image.connector.hdmi.signalrange	138
image.connector.hdmi.detectedsignal	138	image.display.desireddisplaymode	138
image.connector.hdmi.name	139	image.display.frequency	139
image.contrast	140	image.dynamiccontrast.level	140
image.display.displaymode	140	image.gammatype	140
image.display.synchronouslock	141	image.orientation	141
image.gamma	141	image.processing.blacklevel.basicblacklevel.bottom	142
image.intensity	142	image.processing.blacklevel.basicblacklevel.left	142
image.orientationmode	143	image.processing.blacklevel.basicblacklevel.right	143
image.processing.blacklevel.basicblacklevel.enable	143	image.processing.blacklevel.bluegain	144
image.processing.blacklevel.basicblacklevel.level	144	image.processing.blacklevel.file.selected	144
image.processing.blacklevel.basicblacklevel.top	144	image.processing.blacklevel.linkrgb	145
image.processing.blacklevel.file.enable	145	image.processing.blend.basicblend.bottom	146

Pulse API		Reference Guide	Version 2.5
image.processing.blacklevel.greengain	146	image.processing.blend.basicblend.left	146
image.processing.blacklevel.redgain	147	image.processing.blend.basicblend.top	147
image.processing.blend.basicblend.enable	148	image.processing.blend.file.maxselected	148
image.processing.blend.basicblend.right	148	image.processing.blend.scurve	149
image.processing.blend.file.enable	149	image.processing.tiltedbrightness.horizontalcorrection	149
image.processing.blend.file.selected	150	image.processing.transportdelay.actual	150
image.processing.tiltedbrightness.enable	150	image.processing.transportdelay.minimum	151
image.processing.tiltedbrightness.verticalcorrection	151	image.processing.warp.bow.bottomleftv	152
image.processing.transportdelay.desired	152	image.processing.warp.bow.bottomrightv	152
image.processing.warp.bow.bottomleftu	152	image.processing.warp.bow.symmetric	153
image.processing.warp.bow.bottomrightu	153	image.processing.warp.bow.topleftv	153
image.processing.warp.bow.enable	153	image.processing.warp.bow.toprightv	153
image.processing.warp.bow.topleftu	154	image.processing.warp.file.enable	154
image.processing.warp.bow.toprightu	154	image.processing.warp.fourcorners.bottomleft	154
image.processing.warp.enable	155	image.processing.warp.fourcorners.enable	155
image.processing.warp.file.selected	155	image.processing.warp.fourcorners.screenwidth	155
image.processing.warp.fourcorners.bottomright	156	image.processing.warp.fourcorners.topright	156
image.processing.warp.fourcorners.screenheight	156	image.processing.warp.position.horizontal	156
image.processing.warp.fourcorners.topleft	156	image.processing.warp.warpstatus	157
image.processing.warp.position.enable	157	image.processing.warp.zoom.enable	157
image.processing.warp.position.vertical	158	image.processing.warp.zoom.size	158
image.processing.warp.warpstatusdescription	158	image.resolution.availableresolutions	158
image.processing.warp.zoom.factor	159	image.resolution.display.size	159
image.resolution.alpha.size	159	image.resolution.output.size	159
image.resolution.beta.size	159	image.resolution.resolution	160
image.resolution.osd.size	160	image.saturation	160
image.resolution.processing.size	160	image.source.displayport.capabilities	161
image.resolution.size	161	image.source.displayport.name	161
image.sharpness	161	image.source.hdmi.layout	161
image.source.displayport.layout	162	image.stereo.darktime	162
image.source.hdmi.capabilities	162	image.stereo.glassync.delaymaximum	162
image.source.hdmi.name	163	image.stereo.glassync.invert	163
image.stereo.glassync.delay	163	image.testpattern.available	163
image.stereo.glassync.delayminimum	164	image.testpattern.iterator.calibration.list	164
image.stereo.swapframepair	164	image.testpattern.iterator.iris.list	165
image.testpattern.enablefilemanagement	165	image.testpattern.iterator.optics.list	165
image.testpattern.iterator.calibration.size	165	image.testpattern.iterator.shift.list	166
image.testpattern.iterator.iris.size	166	image.testpattern.iterator.stereo.list	166
image.testpattern.iterator.optics.size	166	image.testpattern.iterator.xpr.list	167
image.testpattern.iterator.shift.size	167	image.testpattern.nosignal.pattern	167
image.testpattern.iterator.stereo.size	167	image.testpattern.show	167
image.testpattern.iterator.xpr.size	168	image.window.main.position	168
image.testpattern.selected	168	image.window.main.size	168
image.window.main.activesignal	169	image.window.main.sourceinvalidityreasons	169

Pulse API		Reference Guide	Version 2.5
image.window.main.scalingmode	169	iot.certificate.available	169
image.window.main.source	170	iot.certificate.valid	170
iot.certificate.applicable	170	iot.certificate.validitystart	170
iot.certificate.issuer	171	iot.cloudservices.enable	171
iot.certificate.validityend	171	iot.proxy.enabled	171
iot.cloudservices.connectionstate	172	iot.proxy.port	172
iot.proxy.address	173	iot.requiretermagreement	173
iot.proxy.password	173	keydispatcher.repeatdelay	173
iot.proxy.username	173	network.device.lan.carrier	174
iot.termagreement	174	network.device.lan.devicetype	174
keydispatcher.repeatinterval	174	network.device.lan.ip4config	175
network.device.lan.configuration	175	network.device.lan.ip6config	175
network.device.lan.hwaddress	175	network.device.lan.speed	176
network.device.lan.ip4configmanual	176	network.device.lan.stateinfo	176
network.device.lan.ip6configmanual	176	network.hostname	177
network.device.lan.state	177	notification.count	177
network.domain	177	optics.focus.enabled	178
network.version	178	optics.focus.limits.reverse	178
optics.focus.calibrationstate	178	optics.focus.minposition	179
optics.focus.limits.forward	179	optics.focus.safetocalibrate	179
optics.focus.maxposition	179	optics.focus.state	180
optics.focus.position	180	optics.irisoperation	180
optics.focus.safetoooperate	180	optics.lens.lens	181
optics.focus.target	181	optics.lens.position.list	181
optics.irisoperationavailable	181	optics.lens.position.test1.focus	182
optics.lens.metadata	182	optics.lens.position.test1.shifthorizontal	182
optics.lens.position.selected	182	optics.lens.position.test1.status	183
optics.lens.position.test1.name	183	optics.lens.throwratio	183
optics.lens.position.test1.shiftvertical	183	optics.lensshift.horizontal.calibrationstate	184
optics.lens.position.test1.zoom	184	optics.lensshift.horizontal.limits.forward	184
optics.lenspresent	184	optics.lensshift.horizontal.maxposition	184
optics.lensshift.horizontal.enabled	184	optics.lensshift.horizontal.position	185
optics.lensshift.horizontal.limits.reverse	185	optics.lensshift.horizontal.safetoooperate	185
optics.lensshift.horizontal.minposition	185	optics.lensshift.horizontal.target	185
optics.lensshift.horizontal.safetocalibrate	186	optics.lensshift.vertical.enabled	186
optics.lensshift.horizontal.state	186	optics.lensshift.vertical.limits.reverse	186
optics.lensshift.vertical.calibrationstate	187	optics.lensshift.vertical.minposition	187
optics.lensshift.vertical.limits.forward	187	optics.lensshift.vertical.safetocalibrate	187
optics.lensshift.vertical.maxposition	188	optics.lensshift.vertical.state	188
optics.lensshift.vertical.position	189	optics.shutter.position	189
optics.lensshift.vertical.safetoooperate	189	optics.zoom.calibrationstate	189
optics.lensshift.vertical.target	190	optics.zoom.limits.forward	190
optics.shutter.target	190	optics.zoom.maxposition	191
optics.zoom.enabled	191	optics.zoom.position	191

Pulse API		Reference Guide	Version 2.5
optics.zoom.limits.reverse	191	optics.zoom.safetoperate	192
optics.zoom.minposition	192	optics.zoom.target	192
optics.zoom.safetocalibrate	192	profile.presetassignments	192
optics.zoom.state	193	property.notifyself	193
profile.domains	193	protocols.pjlink.enable	193
profile.profiles	193	remotecontrol.broadcastaddress	194
protocols.pjlink.authenticationrequired	194	remotecontrol.sensors.front.name	194
remotecontrol.address	195	remotecontrol.sensors.rear.name	195
remotecontrol.sensors.front.enable	195	remotecontrol.sensors.side.name	195
remotecontrol.sensors.rear.enable	195	remotecontrol.sensors.wiredremote-layer1.name	196
remotecontrol.sensors.side.enable	196	remotecontrol.sensors.wiredremote.name	196
remotecontrol.sensors.wiredremote-layer1.enable	196	scheduler.actiontypes	196
remotecontrol.sensors.wiredremote.enable	197	scheduler.pause	197
scheduler.actions	197	scheduler.transitions	198
scheduler.nextscheduledaction	198	screen.luminance	198
scheduler.scheduledactions	198	screen.screenize	199
screen.hdrboost	199	statistics.laser.runtime.value	199
screen.screenaspect	199	statistics.laser.plate01.bank02.runtimeseconds.value	200
statistics.illumination on uptime.value	200	statistics.laser.plate02.bank02.runtimeseconds.value	200
statistics.laser.plate01.bank01.runtimeseconds.value	200	statistics.maintenance.colorwheel.value	201
statistics.laser.plate02.bank01.runtimeseconds.value	201	statistics.projector on.value	201
statistics.maintenance.actuator.value	202	statistics.projector.value	202
statistics.maintenance.phosphorwheel.value	202	system.colorwheel	202
statistics.projector uptime.value	203	system.date.availabletimezones	203
system.articlenumber	203	system.date.ntp.enabled	203
system.colorwheelname	203	system.date.timezone	204
system.date.ntp.connected	204	system.eco.enable	204
system.date.ntp.server	204	system.firmwareversion	204
system.eco.available	205	system.initialstate	205
system.familyname	205	system.license.available	205
system.health	206	system.license.register.mandatory	206
system.license.applicable	206	system.license.valid	206
system.license.options	206	system.name	207
system.license.register.valid	207	system.on.timeout.enable	207
system.modelname	207	system.operationalmodes.availablemodes	207
system.on.timeout.duration	208	system.ready.timeout.duration	208
system.on.timeout.remaining	208	system.ready.timeout.remaining	208
system.operationalmodes.mode	209	system.resetresult	209
system.ready.timeout.enable	209	system.serialnumber	209
system.resetprogress	210	system.standby.enable	210
system.resetstatus	210	system.standby.timeout.enable	210
system.standby.available	211	system.state	211
system.standby.timeout.duration	211	ui.access.enduser	212
system.standby.timeout.remaining	212	ui.backlight.timeout	212

Pulse API		Reference Guide	Version 2.5
system.targetstate	212	ui.language	212
ui.backlight.state	213	ui.layer.advancedblend.enable	213
ui.hasstealthmode	213	ui.layer.basicblacklevel.color	213
ui.layer.advancedblend.drawing	214	ui.layer.basicblacklevel.selection	214
ui.layer.advancedblend.palette	214	ui.layer.basicblend.color	214
ui.layer.basicblacklevel.enable	215	ui.layer.basicblend.selection	215
ui.layer.basicblacklevel.selectioncolor	215	ui.layer.blank.border	215
ui.layer.basicblend.enable	216	ui.layer.blank.enable	216
ui.layer.basicblend.selectioncolor	216	ui.layer.blank.show	216
ui.layer.blank.bordercolor	217	ui.layer.blank.text	217
ui.layer.blank.icon	217	ui.layer.fourcorner.enable	217
ui.layer.blank.showtext	218	ui.layer.fourcorner.lines	218
ui.layer.fourcorner.cornercolor	218	ui.layer.fourcorner.selectioncolor	218
ui.layer.fourcorner.linecolor	218	ui.layer.grid.enable	219
ui.layer.fourcorner.selection	219	ui.layer.grid.mark	219
ui.layer.grid.color	219	ui.layer.grid.showlines	219
ui.layer.grid.lines	220	ui.length	220
ui.layer.grid.points	220	ui.luminance	220
ui.layer.grid.showpoints	221	ui.menuposition	221
ui.lensmenu	221	ui.messageposition	221
ui.menu	221	ui.osd	222
ui.message	222	ui.poweroffhint	222
ui.minimize	222	ui.sensors.airpressure.value	223
ui.patternmenu	223	ui.sensors.ambienttemperature.value	223
ui.sensors.airpressure.unit	223	ui.sensors.humidity.value	223
ui.sensors.ambienttemperature.unit	224	ui.sensors.mainsvoltage.value	224
ui.sensors.humidity.unit	224	ui.sourcemenu	224
ui.sensors.mainsvoltage.unit		ui.sourcesignalposition	
ui.showmessage		ui.splashscreen.timeoutseconds	
ui.sourcesignal		ui.temperature	
ui.splashscreen.show		ui.touchscreen.mode	
ui.stealthmode		user.admin.activesessioncount	
ui.theme		user.admin.group	
ui.touchscreen.status		user.admin.username	
user.admin.enabled		user.authenticationrequired	
user.admin.publickey		user.currentuser.activesessioncount	
user.admin.userslug		user.currentuser.group	
user.availablegroups		user.currentuser.username	
user.currentuser.enabled		user.list	
user.currentuser.publickey		user.poweruser.activesessioncount	
user.currentuser.userslug		user.poweruser.group	
user.pincodauthenticationenabled		user.poweruser.username	
user.poweruser.enabled		user.resetadministratorchallenge	
user.poweruser.publickey		user.testuser.enabled	

user.poweruser.userslug	user.testuser.publickey
user.testuser.activesessioncount	user.testuser.userslug
user.testuser.group	user.user.enabled
user.testuser.username	user.user.publickey
user.user.activesessioncount	user.user.userslug
user.user.group	
user.user.username	

Method index

.authenticate	225	.generatenewtoken	225
.getchallenge	225	.introspect	225
.login	226	.logout	226
dmx.listchannels	226	dmx.listmodes	226
environment.getalarminfo	227	environment.getcontrolblocks	227
gpio.gpo.trigger1.listavailablemodes	229	gpio.gpo.trigger1.pulse	229
gpio.gpo.trigger2.listavailablemodes	229	gpio.gpo.trigger2.pulse	230
gpio.listinputs	230	gpio.listoutputs	230
illumination.clo.engage	230	image.brilliantcolor.list	230
image.color.p7.custom.copypresettocustom	231	image.color.p7.custom.makebalanced	231
image.color.p7.custom.resetpreset	231	image.color.p7.custom.resettonative	231
image.color.rgbmode.nextrgbmode	231	image.connector.displayport.edid.list	232
image.connector.hdmi.edid.list	232	image.connector.list	232
image.display.listdisplaymodes	233	image.list	233
image.listgammatypes	233	image.processing.blacklevel.basicblacklevel.getblacklevelarea	233
image.processing.blacklevel.basicblacklevel.getwarpedblacklevelarea	236	image.processing.blacklevel.file.delete	239
image.processing.blacklevel.file.list	240	image.processing.blend.basicblend.getblendarea	240
image.processing.blend.basicblend.getwarpedblendarea	245	image.processing.blend.file.delete	251
image.processing.blend.file.list	251	image.processing.warp.file.delete	251
image.processing.warp.file.list	251	image.processing.warp.fourcorners.getscaledcorners	251
image.processing.warp.warpscaledpoints	253	image.processing.warpgrid.getgrid	254
image.processing.warpgrid.getgridsize	255	image.processing.warpgrid.getscaledgrid	255
image.resolution.list	256	image.source.displayport.listconnectors	256
image.source.hdmi.listconnectors	257	image.source.list	258
image.stereo.listdarktime	258	image.testpattern.file.delete	259
image.testpattern.file.list	259	image.testpattern.getproperties	259
image.testpattern.iterator.calibration.listpossiblepatterns	260	image.testpattern.iterator.calibration.next	260
image.testpattern.iterator.iris.listpossiblepatterns	260	image.testpattern.iterator.iris.next	260
image.testpattern.iterator.optics.listpossiblepatterns	260	image.testpattern.iterator.optics.next	261
image.testpattern.iterator.shift.listpossiblepatterns	261	image.testpattern.iterator.shift.next	261
image.testpattern.iterator.stereo.listpossiblepatterns	261	image.testpattern.iterator.stereo.next	261
image.testpattern.iterator.xpr.listpossiblepatterns	262	image.testpattern.iterator.xpr.next	262
image.testpattern.list	262	image.testpattern.listinternal	263
image.testpattern.nosignal.list	263	image.testpattern.setproperties	263
image.testpattern.toggle	264	image.window.list	264
keydispatcher.postevent	264	led.activity	266
led.list	266	led.muteall	266
lightmeasurement.getlightoutput	266	network.list	267
notification.caution.list	267	notification.dismiss	268
notification.error.list	268	notification.info.list	269
notification.list	270	notification.listsuppressed	271
notification.log	271	notification.suppress	272

Pulse API	Reference Guide		Version 2.5
notification.unsuppress	272	notification.unsuppressall	273
notification.warning.list	273	optics.focus.calibrate	274
optics.focus.runforward	274	optics.focus.runforwardtime	274
optics.focus.runreverse	274	optics.focus.runreversetime	274
optics.focus.stepforward	275	optics.focus.stepreverse	275
optics.focus.stop	275	optics.lens.list	276
optics.lensshift.horizontal.calibrate	276	optics.lensshift.horizontal.runforward	276
optics.lensshift.horizontal.runforwardtime	276	optics.lensshift.horizontal.runreverse	277
optics.lensshift.horizontal.runreversetime	277	optics.lensshift.horizontal.stepforward	277
optics.lensshift.horizontal.stepreverse	278	optics.lensshift.horizontal.stop	278
optics.lensshift.vertical.calibrate	278	optics.lensshift.vertical.runforward	278
optics.lensshift.vertical.runforwardtime	278	optics.lensshift.vertical.runreverse	279
optics.lensshift.vertical.runreversetime	279	optics.lensshift.vertical.stepforward	279
optics.lensshift.vertical.stepreverse	280	optics.lensshift.vertical.stop	280
optics.shifttocenter	280	optics.shutter.getobjectpath	280
optics.shutter.getstate	281	optics.shutter.toggle	281
optics.zoom.calibrate	281	optics.zoom.runforward	281
optics.zoom.runforwardtime	281	optics.zoom.runreverse	282
optics.zoom.runreversetime	282	optics.zoom.stepforward	282
optics.zoom.stepreverse	283	optics.zoom.stop	283
peripheral.autodetect	283	peripheral.list	283
profile.activatepreset	284	profile.activateprofile	284
profile.assignpreset	284	profile.createprofile	285
profile.deleteprofile	285	profile.getdomainsforprofile	285
profile.presetforprofile	286	profile.profileforpreset	286
remotecontrol.listsensors	286	scheduler.create	287
statistics.illumination on uptime.getname	287	statistics.illumination on uptime.getunit	287
statistics.laser runtime.getname	288	statistics.laser runtime.getunit	288
statistics.laser.plate01.bank01.runtimeseconds.getname	288	statistics.laser.plate01.bank01.runtimeseconds.getunit	288
statistics.laser.plate01.bank02.runtimeseconds.getname	289	statistics.laser.plate01.bank02.runtimeseconds.getunit	289
statistics.laser.plate02.bank01.runtimeseconds.getname	289	statistics.laser.plate02.bank01.runtimeseconds.getunit	290
statistics.laser.plate02.bank02.runtimeseconds.getname	290	statistics.laser.plate02.bank02.runtimeseconds.getunit	290
statistics.listcounters	291	statistics.maintenance.actuator.getname	291
statistics.maintenance.actuator.getunit	292	statistics.maintenance.colorwheel.getname	292
statistics.maintenance.colorwheel.getunit	292	statistics.maintenance.phosphorwheel.getname	293
statistics.maintenance.phosphorwheel.getunit	293	statistics.projector on.getname	293
statistics.projector on.getunit	293	statistics.projector uptime.getname	294
statistics.projector uptime.getunit	294	statistics.projector.getname	295
statistics.projector.getunit	295	system.activity	295
system.boards.getboardlist	295	system.boards.getmissingboardlist	296
system.date.getlocaldate	296	system.date.getsystemdate	297
system.getidentifications	298	system.getsystemdate	298
system.gotoeco	299	system.gotoready	299
system.license.file.saveregistrationtousb	299	system.license.file.uploadfromusb	300

Pulse API	Reference Guide		Version 2.5
system.listresetdomains	300	system.poweroff	301
system.poweron	301	system.reset	301
system.resetall	302	ui.leds.setstate	304
ui.settings.get	304	ui.settings.getfonticons	304
ui.settings.geticons	305	ui.settings.keys	305
ui.settings.list	306	ui.settings.remove	306
ui.settings.set	306	ui.togglelensmenu	307
ui.togglemenu	307	ui.toggleosd	307
ui.togglepatternmenu	307	ui.togglesourcemenu	307
ui.togglestealthmode	307	ui.touchscreen.tapped	308
user.admin.changegroup	308	user.admin.changepassword	308
user.admin.changepincode	309	user.admin.changepublickey	309
user.admin.changeusername	309	user.admin.disable	309
user.admin.enable	310	user.admin.invalidatetokens	310
user.admin.logouteverywhere	310	user.admin.removepassword	310
user.admin.removepincode	310	user.currentuser.changegroup	310
user.currentuser.changepassword	311	user.currentuser.changepincode	311
user.currentuser.changepublickey	311	user.currentuser.changeusername	311
user.currentuser.disable	312	user.currentuser.enable	312
user.currentuser.invalidatetokens	312	user.currentuser.logouteverywhere	312
user.currentuser.removepassword	312	user.currentuser.removepincode	313
user.poweruser.changegroup	313	user.poweruser.changepassword	313
user.poweruser.changepincode	313	user.poweruser.changepublickey	314
user.poweruser.changeusername	314	user.poweruser.disable	314
user.poweruser.enable	314	user.poweruser.invalidatetokens	315
user.poweruser.logouteverywhere	315	user.poweruser.removepassword	315
user.poweruser.removepincode	315	user.resetadministrator	315
user.testuser.changegroup	316	user.testuser.changepassword	316
user.testuser.changepincode	316	user.testuser.changepublickey	316
user.testuser.changeusername	317	user.testuser.disable	317
user.testuser.enable	317	user.testuser.invalidatetokens	317
user.testuser.logouteverywhere	317	user.testuser.removepassword	318
user.testuser.removepincode	318	user.user.changegroup	318
user.user.changepassword	318	user.user.changepincode	318
user.user.changepublickey	319	user.user.changeusername	319
user.user.disable	319	user.user.enable	320
user.user.invalidatetokens	320	user.user.logouteverywhere	320
user.user.removepassword	320	user.user.removepincode	320

Signal index

loggedout	321	modelupdated	321
dmx.shutdownrequest	322	image.brilliantcolor.listchanged	322
image.connector.displayport.edid.listchanged	322	image.connector.hdmi.edid.listchanged	322
image.processing.blacklevel.file.listchanged	322	image.processing.blend.file.listchanged	322
image.processing.warp.file.listchanged	323	image.processing.warpgrid.changed	323
image.processing.warpgrid.gridchanged	323	image.testpattern.added	323
image.testpattern.changed	324	image.testpattern.file.listchanged	324
image.testpattern.removed	324	keydispatcher.keyevent	325
network.added	325	network.removed	325
notification.caution.dismissed	325	notification.caution.emitted	325
notification.dismissed	325	notification.emitted	326
notification.error.dismissed	326	notification.error.emitted	327
notification.info.dismissed	327	notification.info.emitted	327
notification.warning.dismissed	327	notification.warning.emitted	327
profile.profileactivated	327	system.date.systemdatechanged	327
system.identificationchanged	328	system.license.licensechanged	328
system.performed	328	ui.settings.added	329
ui.settings.changed	329	ui.settings.removed	329

File index

bio.logger.transfer	330	image.connector.edid.transfer	330
image.processing.blacklevel.file.transfer	331	image.processing.blend.file.transfer	331
image.processing.warp.file.transfer	331	image.testpattern.file.transfer	332
notification.logger.transfer	332	system.license.file.registration	333
ui.capture.lcd	333	ui.capture.osd	334
ui.splashscreen.file	334		