Osprey® Talon Decoder Configuration and Control API User Guide

Contents

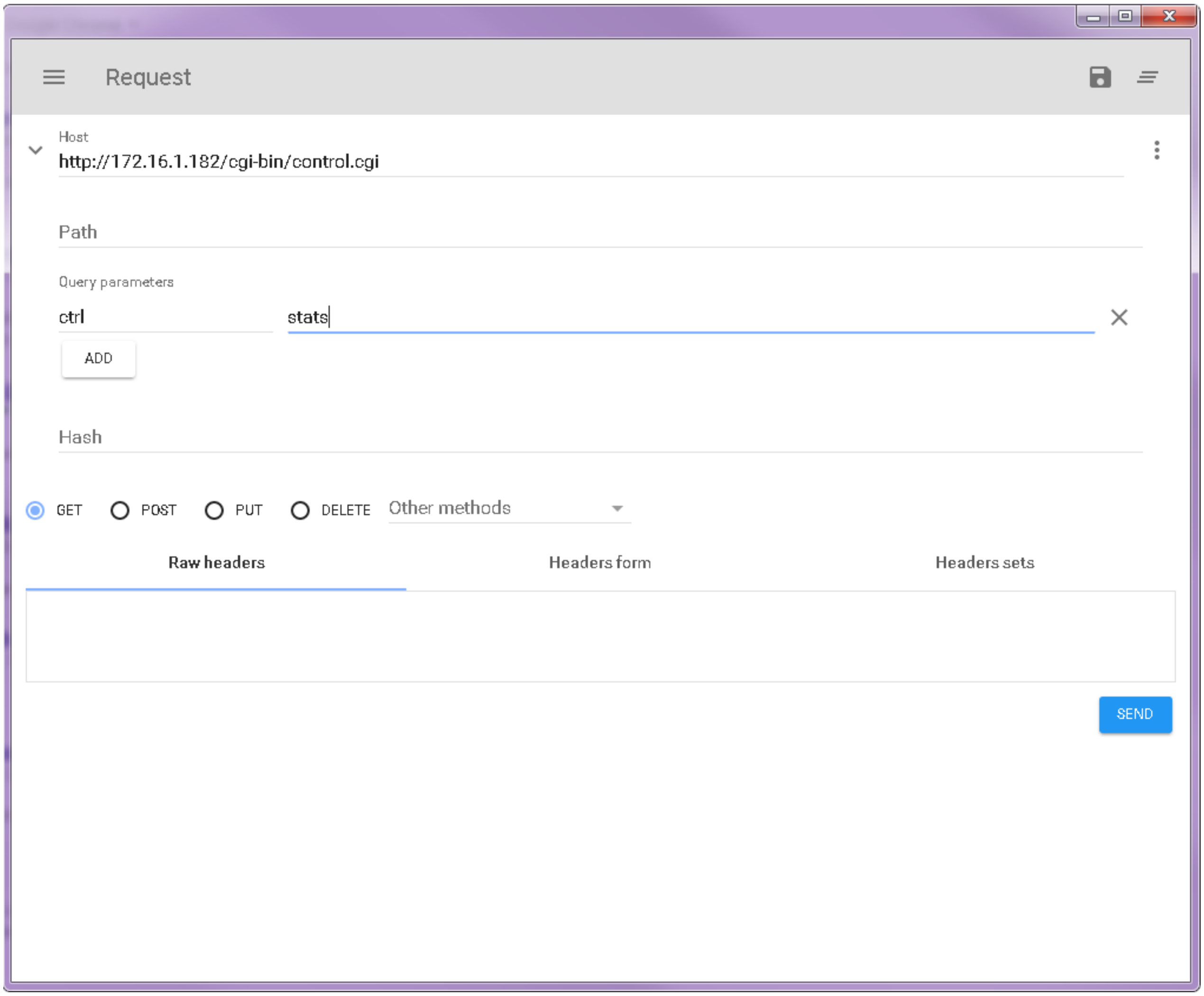
Description

This document describes the Osprey Talon Configuration and Control API (OTCC API). The API uses HTTP PUT/GET requests and values are returned through JSON.

All Osprey Talon Decoders are shipped with the OTCC API pre-installed. This API can be used in C++, C#, Java, IOS, etc.

Testing

The functions in this document can all be tested outside of a development environment using CURL in Linux or the Advanced REST Client (ARC) that is free for Chrome. An example is provided below for a GET to obtain the stats from the Osprey Talon.



Request

Host
http://172.16.1.182/cgi-bin/control.cgi

Path

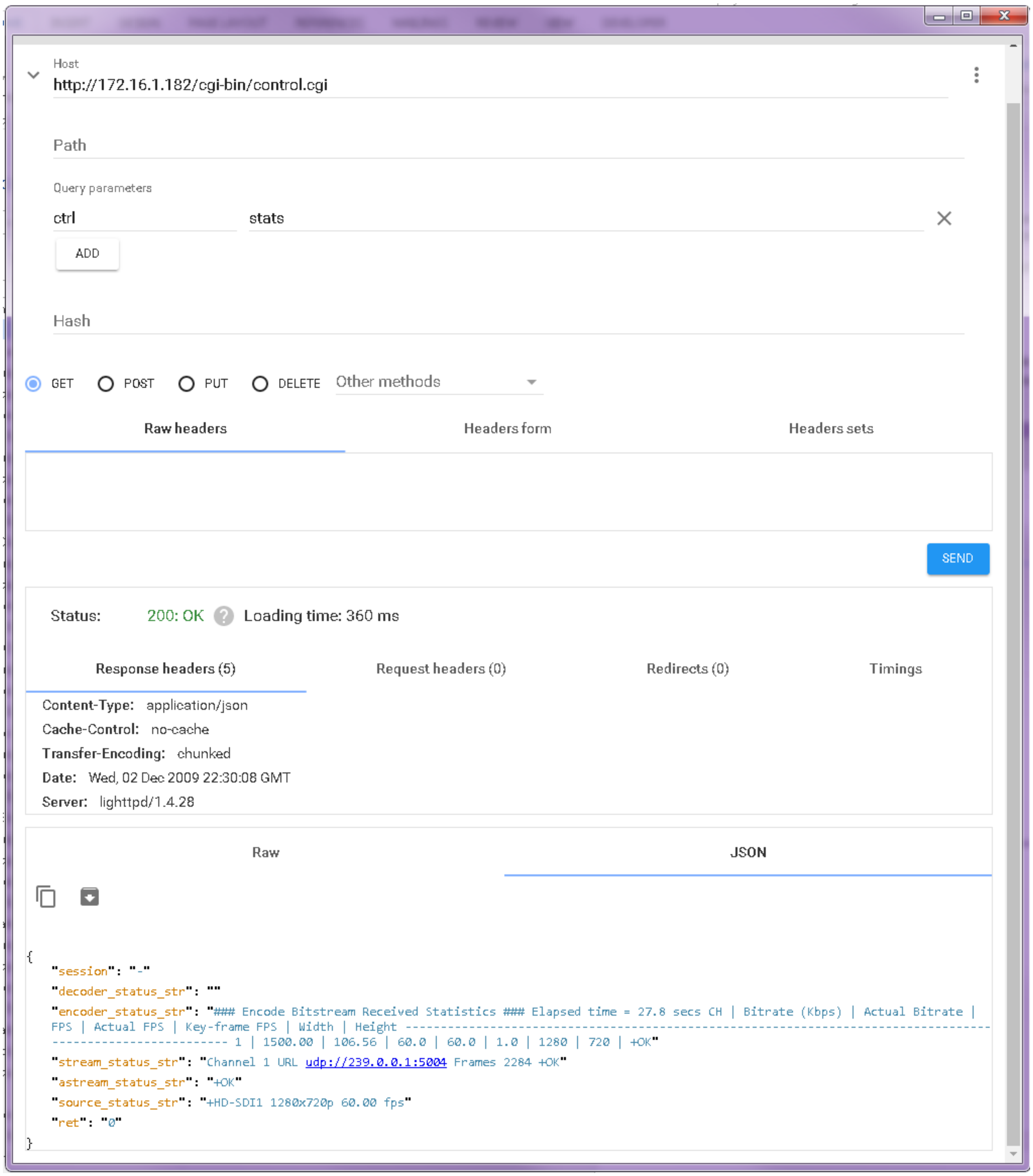
Query parameters

ctrl	stats

Hash

● GET ○ POST ○ PUT ○ DELETE Other methods

The server response headers and JSON results are all shown in the ARC windows.



Host
http://172.16.1.182/cgi-bin/control.cgi

Query parameters

ctrl	stats

● GET ○ POST ○ PUT ○ DELETE

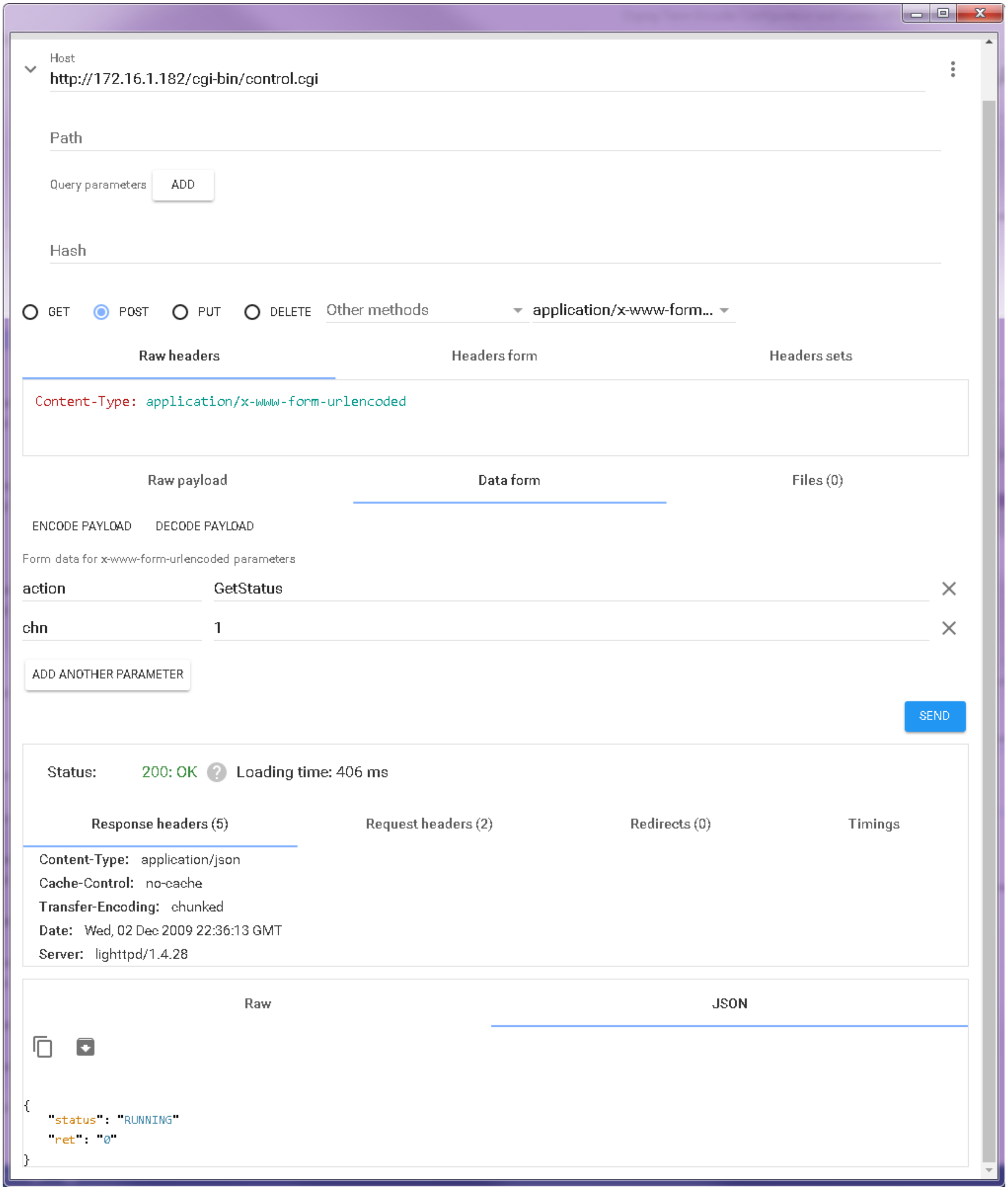
Status: 200: OK Loading time: 360 ms

Response headers (5)

Content-Type: application/json
Cache-Control: no-cache
Transfer-Encoding: chunked
Date: Wed, 02 Dec 2009 22:30:08 GMT
Server: lighttpd/1.4.28

```
{
  "session": "-"
  "decoder_status_str": ""
  "encoder_status_str": "### Encode Bitstream Received Statistics ### Elapsed time = 27.8 secs CH | Bitrate (Kbps) | Actual Bitrate |
  FPS | Actual FPS | Key-frame FPS | Width | Height -----------------------------------------------------------------------------------------------
  ------------------------ 1 | 1500.00 | 106.56 | 60.0 | 60.0 | 1.0 | 1280 | 720 | +OK"
  "stream_status_str": "Channel 1 URL udp://239.0.0.1:5004 Frames 2284 +OK"
  "astream_status_str": "+OK"
  "source_status_str": "+HD-SDI1 1280x720p 60.00 fps"
  "ret": "0"
}
```

Similarly the ARC can be used to test the POST functions such as the GetStatus function is in this example.



Host
http://172.16.1.182/cgi-bin/control.cgi

○ GET ● POST ○ PUT ○ DELETE Other methods application/x-www-form...

```
Content-Type: application/x-www-form-urlencoded
```

Form data for x-www-form-urlencoded parameters

action	GetStatus
chn	1

Status: 200: OK Loading time: 406 ms

Response headers (5)

```
Content-Type:  application/json
Cache-Control:  no-cache
Transfer-Encoding:  chunked
Date:  Wed, 02 Dec 2009 22:36:13 GMT
Server:  lighttpd/1.4.28
```

```
{
  "status": "RUNNING"
  "ret": "0"
}
```

Starting a Decoder

Description

Starts the selected channel on the Talon Decoder.

Method: POST
http://DecoderIP/cgi-bin/control.cgi?action=StartChannel&chn=*{channel}*

Required Parameters

Name	Type	Description	Example
action	*string*	*The start command.*	StartChannel
chn	*string*	*This is a numerical value that represents the channel to start*	1

Response Payload

```
{
"status": "POLLING"
"ret": "0"
}
```

Name	Type	Description	Example
status	*string*	The last status of the decoder. (DECODING, error responses, etc.)	“RUNNING”
ret	*string*	Return Code	“0”

Curl Example

```
$ curl - -X POST http://10.0.0.1/cgi-bin/control.cgi?action=StartChannel&chn=1
```

Stopping an Decoder

Description

Stops the selected channel on the Talon Decoder.

```
Method: POST
http://DecoderIP/cgi-bin/control.cgi?action=StopChannel&chn={channel}
```

Required Parameters

Name	Type	Description	Example
action	*string*	*The stop command.*	StopChannel
chn	*string*	*This is a numerical value that represents the channel to stop*	1

Response Payload

```
{
"status": "POLLING"
"ret": "0"
}
```

Name	Type	Description	Example
status	*string*	The last status of the decoder. (POLLING, STOPPED, error responses, etc.)	“STOPPED”
ret	*string*	Return Code	“0”

Curl Example

```
$ curl - -X POST http://10.0.0.1/cgi-bin/control.cgi?action=StopChannel&chn=1
```

Delete Channel

Description

Deletes the selected channel on the Talon Decoder.

Method: POST
http://DecodeIP/cgi-bin/control.cgi?action=DeleteChannel&chn=*{channel}*

Required Parameters

Name	Type	Description	Example
action	*string*	*The delete channel command.*	DeleteChannel
chn	*string*	*This is a numerical value that represents the channel to delete*	3

Response Payload

```
{
"channels": "C1,C2"
"ret": "0"
}
```

Name	Type	Description	Example
channels	*string*	The list of currently configured channels.	“C1,C2”
ret	*string*	Return Code	“0”

Curl Example

```
$ curl - -X POST http://10.0.0.1/cgi-bin/control.cgi?action=DeleteChannel&chn=3
```

Add Channel

Description

Creates an additional channel with the channel number specified on the Talon Decoder.

```
Method: POST
http://DecoderIP/cgi-bin/control.cgi?action=AddChannel&newchn={channel}
```

Required Parameters

Name	Type	Description	Example
action	*string*	*The add channel command.*	AddChannel
newchn	*string*	*This is a numerical value that represents the channel to add*	1

Response Payload

```
{
"ret": "0"
}
```

Name	Type	Description	Example
ret	*string*	Return Code	“0”

Curl Example

```
$ curl - -X POST http://10.0.0.1/cgi-bin/control.cgi?action=AddChannel&chn=1
```

Clone Channel

Description

Clones a selected channel to a specified channel number on the Talon Decoder.

Method: POST
http://DecoderIP/cgi-bin/control.cgi?action=CloneChannel&from=*{channel}*& from=*{channel2}*

Required Parameters

Name	Type	Description	Example
action	*string*	*The clone channel command.*	CloneChannel
from	*string*	*This is a numerical value that represents the channel to clone*	1
to	*string*	*This is a numerical value that represents the channel to create*	2

Response Payload

```
{
"status": "Idle"
"ret": "0"
}
```

Name	Type	Description	Example
status	*string*	The last status of the decoder. (POLLING, STOPPED, error responses, etc.)	“!Error: Bad destination [10.0.0.1]”
ret	*string*	Return Code	“0”

Curl Example

```
$ curl - -X POST http://10.0.0.1/cgi-bin/control.cgi?action=CloneChannel&chn=1
```

Get Decoder Status

Description

Returns the decoder station of a specified channel number on the Talon Decoder.

```
Method: POST
http://Decoder/cgi-bin/control.cgi?action=GetStatus&chn={channel}
```

Required Parameters

Name	Type	Description	Example
action	*string*	*The get status command.*	GetStatus
chn	*string*	*This is a numerical value that represents the channel to stop*	1

Response Payload

```
{
"status": "Idle"
"ret": "0"
}
```

Name	Type	Description	Example
status	*string*	The last status of the decoder. (STOPPED, error responses, etc.)	“!Error: Bad destination [10.0.0.1]”
ret	*string*	Return Code	“0”

Curl Example

```
$ curl - -X POST http://10.0.0.1/cgi-bin/control.cgi?action=GetStatus&chn=1
```

Get Stats

Method: GET
http://DecoderIPIP/cgi-bin/control.cgi

Required Query Parameters

Name	Type	Description	Example
ctrl	*string*	*The get stats command.*	stats

**This method returns the last known status for each item. To update the status, first call individual methods: EncoderStatus, SourceStatus, AStreamStatus, StreamStatus prior to calling the status function.*

Response Payload

```
{

"session": "-"

"decoder_status_str": "No decodes currently running"

"encoder_status_str": "### Encode Bitstream Received Statistics ### Elapsed
time = 27.8 secs CH | Bitrate (Kbps) | Actual Bitrate | FPS | Actual FPS |
Key-frame FPS | Width | Height ----------------------------------------------
-------------------------------------------------------------- 1 | 1500.00 |
106.56 | 60.0 | 60.0 | 1.0 | 1280 | 720 | +OK"

"stream_status_str": "Channel 1 URL udp://239.0.0.1:5004 Frames 2284 +OK"

"astream_status_str": "+OK"

"source_status_str": "+HD-SDI1 1280x720p 60.00 fps"

"ret": "0"

}
```

Name	Type	Description	Example
session	*string*	The last status of the decoder. (POLLING, STOPPED, error responses, etc.)	“-“
decoder_status_str	*string*	Decoder status for all channels	N/A

encoder_status_str	*string*	Encoder status for all channels	Ignore. Valid on encoders only.
stream_status_str	*string*	Return Code	"Channel 1 URL Frames 2284 +OK"
astream_status_str	*string*	The last status of the encoder. (POLLING, STOPPED, error responses, etc.)	Ignore. Valid on encoders only.
source_status_str	*string*	Return Code	Ignore. Valid on encoders only.
ret	*string*	Return Code	“0”

Curl Example

```
$ curl - -X GET http://10.0.0.1/cgi-bin/control.cgi?ctrl=stats
```

Restart Talon

Description

Restarts the Osprey Talon device immediately.

```
Method: POST
http://DecoderIP/cgi-bin/control.cgi?action=RestartBoard
```

Required Parameters

Name	Type	Description	Example
action	*string*	*The restart command.*	RestartBoard

Response Payload

```
No response.  System immediately restarts.
```

Curl Example

```
$ curl - -X POST http://10.0.0.1/cgi-bin/control.cgi?action=RestartBoard
```

Modifying an Decoder Property

Description

Modifies selected properties on the Osprey Talon Decoder

Method: POST
https://DecoderIP/cgi-bin/control.cgi?*{property}*=*{value}*

Required Parameters

Name	Type	Description	Example
property	*string*	*The property to change formatted as C{#}_PropertyName where "#" is the channel number of the stream to modify (1, 2, 3)* *(Refer to table 1 for property names and values)*	C1_dec_port

Response Payload

```
{
   "ret": "0"
}
```

Name	Type	Description	Example
ret	*string*	Return Code	"0"

Curl Example

```
$ curl - -X POST https://10.0.0.1/cgi-bin/control.cgi?action=C1_dec_port=5001
```

Table 1: Decoder Properties

Property Name	Valid Values	Description
dec_autostart	0 1	A value of 0 disables autostart. A value of 1 enables autostart.
dec_size	Output size (example 720x480)	Output size resolution
dec_display	"hdmi" – only valid value	Output mode
dec_ip	239.0.0.1	The input URL for the decoder. If using multicast, this will be the multicast address.
dec_port	Numerical Value	The input port to be used for the selected decoding.
dec_filename	String Value	This is the file location for decoding from a file.
dec_audonoff	on off	Enables or disables the audio output for decoded stream.
dec_loop	on off	Enables or disables the looping the input file for playback.

www.ospreyvideo.com