

# Ultra Encode

## Overview

<a href="#">Introduction</a>	1.1
<a href="#">API Agreement</a>	1.2
<a href="#">API Status Codes</a>	1.3
<a href="#">Device Status Mask</a>	1.4
<a href="#">Device Discovery Protocol</a>	1.5
<a href="#">DEMO: Node.js</a>	1.6
<a href="#">DEMO: C</a>	1.7

## Universal Interfaces

<a href="#">get-info</a>	2.1
<a href="#">get-status</a>	2.2
<a href="#">get-settings</a>	2.3
<a href="#">ping</a>	2.4
<a href="#">get-signal-info</a>	2.5
<a href="#">get-report</a>	2.6
<a href="#">export-report</a>	2.7

## General Settings

<a href="#">set-name</a>	3.1
<a href="#">set-first-over</a>	3.2
<a href="#">set-softap</a>	3.3
<a href="#">set-ssdp</a>	3.4
<a href="#">set-volume</a>	3.5
<a href="#">set-video-color</a>	3.6
<a href="#">set-video-input-format</a>	3.7
<a href="#">set-video-output-format</a>	3.8
<a href="#">set-sync-offset</a>	3.9

## Streaming Server

<a href="#">add-server</a>	4.1
<a href="#">enable-server</a>	4.2
<a href="#">set-server</a>	4.3
<a href="#">del-server</a>	4.4
<a href="#">start-test-server</a>	4.5
<a href="#">stop-test-server</a>	4.6
<a href="#">clear-test-server</a>	4.7
<a href="#">get-ndi-sources</a>	4.8

## Encoding Parameters

<a href="#">set-video</a>	5.1
<a href="#">set-audio</a>	5.2

## Reset & Reboot

<a href="#">reboot</a>	6.1
<a href="#">reset-all-settings</a>	6.2

## Log In/Out

<a href="#">login</a>	7.1
<a href="#">logout</a>	7.2

## User Management

<a href="#">get-users</a>	8.1
<a href="#">add-user</a>	8.2
<a href="#">del-user</a>	8.3
<a href="#">ch-password</a>	8.4
<a href="#">set-password</a>	8.5

## Network Settings

<a href="#">set-eth</a>	9.1
<a href="#">scan-wifi-results</a>	9.2
<a href="#">connect-wifi-first</a>	9.3
<a href="#">connect-wifi</a>	9.4
<a href="#">cancel-connect-wifi</a>	9.5
<a href="#">disconnect-wifi</a>	9.6
<a href="#">forget-wifi</a>	9.7
<a href="#">set-connect-wifi-auto</a>	9.8
<a href="#">clear-connect-wifi</a>	9.9
<a href="#">open-softap</a>	9.10
<a href="#">close-softap</a>	9.11

## Firmware Update

<a href="#">update</a>	10.1
<a href="#">upload-update-file</a>	10.2
<a href="#">cancel-download</a>	10.3
<a href="#">online-update-check</a>	10.4
<a href="#">clear-upgrade</a>	10.5
<a href="#">clear-check-update</a>	10.6
<a href="#">set-enable-check-update</a>	10.7

# Introduction

A rich set of APIs are provided for developers to interact with Ultra Encode devices, such as obtaining basic device information including device name, firmware version and etc., modifying device configurations and upgrading firmware. These APIs are based on the HTTP protocol and provide lightweight, connectionless services that return JSON data. With this document, you can acquire a thorough understanding of each API's functions and request method.

APIs in this document apply to:

- Ultra Encode HDMI
- Ultra Encode SDI

# API Agreement

## Overview

- Request protocol: HTTP
- Request method: by default, GET is used to request and commit data, and POST is used to upload a file.
- Request URL: `http://IP/usapi?method=xx&param1=value1&param2=value2...`
- Return data format: when the status code is 200, it returns JSON data, otherwise it returns HTTP status codes.
- Login authentication: carry `sid=xxxxxxxx` in cookies

## Example Response

The JSON formatted data is as follows. The result attribute indicates an [API Status Codes](#). The status 0 indicates a successful request, otherwise the request is failed.

```
{
  "result": 0,
  "cur-status": 65552,
  "last-rec-status": 0,
  "cur-time": 0,
  "box-name": "Ultra Encode A304200908001",
  ...
}
```

## API Status Code

```
{
  retSendWaiting      = 31,      // Reserved
  retLivingAuthErr    = 30,      // Live stream status: authentication error
  retLivingNotset     = 29,      // Live stream address not set
  retLivingDNS        = 28,      // Live stream status: Resolving DNS
  retInit             = 27,      // Initialization
  retLivingAuthing    = 25,      // Live stream status: authorizing
  retLivingWaiting    = 24,      // Live stream status: waiting for connection
  retLivingConnecting = 23,      // Live stream status: connecting to the streaming destination
  retLivingConnected  = 22,      // Live stream status: stream server connected
  retPushReboot       = 21,
  retAudioSignalChange = 20,
  retBlueWrite        = 19,
  retBlueRead         = 18,
  retBlueShutDown     = 17,
  retDiskOn           = 16,
  retDiskOff          = 15,
  retDiskChange       = 14,
  retSnapshotOver     = 13,
  retPushReset        = 12,
  retPushLiving       = 11,
  retPushRecord       = 10,
  retSignalChange     = 9,
  retRouteChange      = 8,
  retIPChange         = 7,
  retNetChange        = 6,
  retCancel           = 5,      // Request canceled
  retLowSpace         = 4,      // Reserved
  retLowSpeed         = 3,      // Reserved
  retRunning          = 2,      // Request is being processed
  retRepeat           = 1,      // Repeated request
  retSucceed          = 0,      // Request success
  errPasswd           = -1,      // Wrong password
  errOccupied         = -2,      // Device occupied
  errDisconnect       = -3,      // Reserved
  errDevice           = -4,
  errDisk             = -5,
  errUnconnect        = -6,
  errKey              = -7,
  errVersion          = -8,
  errBusy             = -9,      // System busy
  errParam            = -10,     // Incorrect request parameters
  errUsage            = -11,     // Reserved
  errTimeout          = -12,
  errIP               = -13,     // Reserved
  errNotFound         = -14,     // Data not found
  errFile             = -15,     // File error
  errNoSpace          = -16,     // Reserved
  errNeedAuth         = -17,     // Unauthorized log-in
  errSystem           = -18,     // System error
  errDiskSpeed        = -19,
  errEmpty            = -20,
  errNetwork          = -21,
  errEvent            = -22,
  errCodec            = -23,
  errBlue             = -24,
  errNoUser           = -25,     // User not exist
  errNoPermissin     = -26,
  errSameName         = -27,     // Name already in use
  errString           = -28,     // Invalid input characters
  errChannelsLimited = -29,     // Streaming 6 sessions simultaneously at most.
  err8MLimited        = -30,     // Reserved
  errFacebookLimited = -31,     // Reserved
  errCodecLimited     = -32,     // Reserved
  err4GLimited        = -33,     // Reserved
  errMWFUnsupported  = -34,     // Update package does not match current model or hardware version
  errNoSignal         = -35,     // No signal
  errSDCard           = -36,
```

```
errXinYueServer      = -37,      // Reserved
errAliYunOSS         = -38,      // Reserved
errSDNoSpace         = -39,      // Reserved
errSDNoPermission    = -40,      // Reserved
errRTSPLimited       = -41,      // Only one RTSP session is supported at a time
errRTSP8MLimited     = -42,      // Reserved
errBandwidthLimited  = -43,      // Reserved
errPortLimited       = -44,      // Stream port occupied
errNDILimited        = -45,      // Only one NDI|HX session is supported at a time
errSRTLimited        = -46,      // Only one SRT Listener session is supported at a time
errNDISettings       = -47,      // The substream can be up to 640x480@60 for live streaming a NDI|HX s
ession
  errSubStreamSettings = -48,      // The substream can be up to 1280x720@30 for live streaming a non-NDI
|HX session
  errHLSLimited       = -49,      // Only one HLS session is supported at a time
  errProtocollimited  = -50,      // Allow simultaneous sessions over the same streaming protocol
}
```

## Device Status Mask

```
{
  statusFirst      = 0x01,      // First booting
  statusRecord     = 0x02,      // Reserved
  statusLiving     = 0x04,      // Reserved
  statusStream     = 0x08,      // Reserved
  statusDiskReady  = 0x10,      // Reserved
  statusRTMPReady  = 0x20,      // Reserved
  statusSoftAP     = 0x40,      // The device has enabled the AP mode
  statusMIC        = 0x100,     // Reserved
  statusPHONE      = 0x200,     // Reserved
  statusOutput     = 0x400,     // Reserved
  statusDiskTest   = 0x1000,    // Reserved
  statusBlue       = 0x2000,    // Reserved
  statusUpgrade    = 0x4000,    // Firmware update is in progress
  statusNetTest    = 0x8000,    // Streaming test is in progress
  statusPasswd     = 0x10000,   // Device password has been set
  statusOccupied   = 0x20000,   // Device has been locked by app(s), at most 2 apps simultaneously
  statusFormatDisk = 0x100000,  // Reserved
  statusFormatSD   = 0x200000,  // Reserved
  statusSearchWifi = 0x400000,  // The device is searching for available Wi-Fi networks
  statusConnectWifi = 0x800000, // The device is connecting to a Wi-Fi network
  statusConnectBlue = 0x1000000, // Reserved
  statusCheckUpgrade = 0x2000000, // The device is checking for new firmware versions
  statusReset      = 0x4000000, // The device is resetting
  stausIPv6        = 0x8000000,  // Reserved
  statusTestLock   = 0x10000000, // Reserved
  statusReboot     = 0x20000000, // The device is rebooting
}
```

# Device Discovery Protocol

You can discover an encode through **multicast protocol** or using **SSDP**.

## Multicast

Obtain the basic configurations and status of the device, including device name, serial number, running status, and network status.

- Multicast address: 239.255.255.250
- Port: 2538

Response Body

```
{
  "version": "1.0",
  "flag": "ssip",
  "product": "Ultra Encode",
  "boxname": "[LH-SDI]-1.2.1",
  "serialnumber": "A304200908001",
  "wifiip": "192.168.48.1",
  "ethip": "192.168.1.217",
  "status": 65600
}
```

Item	Description
version	Indicates the multicast protocol version.
flag	Indicates flag of the multicast protocol.
product	Indicates product type.
boxname	Indicates device name.
serialnumber	Indicates device serial number.
wifiip	Indicates IP address for Wi-Fi.
ethip	Indicates IP address for Ethernet.
status	Indicates <a href="#">Device Status Mask</a> .



# DEMO: Node.js

This chapter introduces how to call the Ultra Encode APIs in Node.js.

Download DEMO: [ultra-encode-api-demo-nodejs.zip](#)

DEMO Structure

```
ultra-encode-api-demo-nodejs
|
|-- httpUtils.js // encapsulates the get and upload method based on the HTTP module of Node.js.
|-- xxxx.mwf // file uploaded by upload.js through invoking the upload-update-file interface.
|-- get.js // requests data using GET
|-- upload.js // uploads file using POST
```

## Requirements

- Operating System: macOS, Linux, or Windows
- Node.js Runtime: LTS releases (8.x and later)

## Running Mode

1. Navigate into the DEMO directory in the terminal.

```
cd ultra-encode-demo-nodejs
```

2. Replace deviceIP with the IP address of the test device in get.js and upload.js.
3. Run get.js.

```
node get
```

4. Run upload.js.

```
node upload
```

# DEMO: C

## Requirements

- Operating System: Windows, macOS, or Linux

## Compilation

- Prepare the curl sdk for your Windows/macOS/Linux OS
- Download DEMO: [ultra-encode-api-demo-c.zip](#)
- Compile "ultra\_encode\_curl.c", and link it to "libcurl"
- Build the ultra\_encode\_curl file

## Example

- Navigate into the bin directory and copy the firmware file here, and run ultra\_encode\_curl.

```
cd ultra-encode-api/demo/c/bin
cp ultra_encode_hdmi_rev_a_1_3_328.mwf linux
cd linux
./ultra_encode_curl <hostip:port>
```

- Sample response

```
***** 1. login *****
login response data:
{
  "result": 0
}

***** 2. get info *****
get info response data:
{
  "result": 0,
  "mac-addr": {
    "eth": "70:b3:d5:75:d0:4c",
    "wifi": "70:b3:d5:75:d0:4d",
    "blue": "70:b3:d5:75:d0:4e"
  },
  "snapshot": "/tmp/sbox-snapshot/sbox-quarter.jpg",
  "product": {
    "sn": "A304201201001",
    "product-id": 772,
    "hardware-ver": "A",
    "firmware-id": 1,
    "firmware-ver-s": "1.3.328",
    "factory-firmware-ver-s": "1.3.328",
    "product-name": "Ultra Encode",
    "module-name": "Ultra Encode HDMI",
    "manu-name": "MAGEWELL",
    "features": 1,
    "max-lock-count": 2
  },
  "audio-range": {
    "hdmi": {
      "max": 6.00,
      "min": -100.00,
      "def": 0.00
    },
    "mic": {
      "max": 55.25,
      "min": -12.00,
      "def": 0.00
    },
    "phone": {
      "max": 6.00,
      "min": -57.00,
```

```
        "def": 0.00
      },
      "codec-cap": {}
    }

**** 3. upload firmware ****
upload firmware response data:
{
  "result": 0,
  "up-to-date": true,
  "version": "1.2.123",
  "size": 12494463
}
```

## get-info

Use this interface to obtain device information, including product and manufacture information, MAC address of the network card, and video and audio parameter value ranges.

### HTTP Request

```
GET http://ip/usapi?method=get-info
```

Parameter	Description
method	get-info

### Response Body

JSON structure is as follows:

```
{
  "result": 0, // returned status
  "mac-addr": {}, // MAC address
  "snapshot": "", // path of snapshot
  "product": {}, // product information
  "audio-range": { // audio parameter value
    "hdm": {},
    "mic": {},
    "phone": {}
  },
  "codec-cap": { // encoding parameter value
    "resolutions": [],
    "durations": [],
    "profile": [],
    "hevc-profile": [],
    "video-kbps": [],
    "audio-kbps": [],
    "gop-sec": [],
    "video-range": [],
    "stat-sec": [],
    "video-codec": []
  },
  "color-range": { // color parameter value
    "contrast": {},
    "brightness": {},
    "saturation": {},
    "hue": {}
  }
}
"video-format": { // video input and output parameter value
  "input-color-fmt": [],
  "output-color-fmt": [],
  "quant-range": [],
  "sat-range": []
}
"living": { // live stream parameter value
  "max-bandwidth": 16384,
  "srt": {
    "latency": [],
    "bandwidth": [],
    "aes": []
  }
}
}
```

### Response Body

```
"result": 0,
  "mac-addr": {
    "eth": "d0:c8:57:80:2d:7c",
```

```

"wifi": "d0:c8:57:80:2d:7d",
"blue": "d0:c8:57:80:2d:7e"
},
"snapshot": "/tmp/sbox-snapshot/sbox-quarter.jpg",
"product": {
  "sn": "A304200731004",
  "product-id": 772,
  "hardware-ver": "A",
  "firmware-id": 1,
  "firmware-ver-s": "1.3.325",
  "factory-firmware-ver-s": "1.3.237",
  "product-name": "Ultra Encode",
  "module-name": "Ultra Encode HDMI",
  "manu-name": "MAGEWELL",
  "features": 1,
  "max-lock-count": 2
},
"audio-range": {
  "spi": {
    "max": 6.00,
    "min": -100.00,
    "def": 0.00
  },
  "linein": {
    "max": 6.00,
    "min": -100.00,
    "def": 0.00
  },
  "lineout": {
    "max": 6.00,
    "min": -57.00,
    "def": 0.00
  }
},
"codec-cap": {
  "resolutions": [
    {
      "w": 480,
      "h": 270
    },
    {
      "w": 480,
      "h": 360
    },
    {
      "w": 640,
      "h": 360
    },
    {
      "w": 640,
      "h": 480
    },
    {
      "w": 720,
      "h": 540
    },
    {
      "w": 720,
      "h": 576
    },
    {
      "w": 768,
      "h": 576
    },
    {
      "w": 800,
      "h": 600
    },
    {
      "w": 960,
      "h": 540
    }
  ]
}

```

```

        "w": 1024,
        "h": 768
    },
    {
        "w": 1280,
        "h": 720
    },
    {
        "w": 1280,
        "h": 800
    },
    {
        "w": 1280,
        "h": 960
    },
    {
        "w": 1280,
        "h": 1024
    },
    {
        "w": 1440,
        "h": 900
    },
    {
        "w": 1440,
        "h": 1080
    },
    {
        "w": 1600,
        "h": 1200
    },
    {
        "w": 1920,
        "h": 1080
    }
],
"durations": [
    {
        "name": "5 FPS",
        "value": 2000000
    },
    {
        "name": "10 FPS",
        "value": 1000000
    },
    {
        "name": "15 FPS",
        "value": 666667
    },
    {
        "name": "24 FPS",
        "value": 416667
    },
    {
        "name": "25 FPS",
        "value": 400000
    },
    {
        "name": "29.97 FPS",
        "value": 333667
    },
    {
        "name": "30 FPS",
        "value": 333333
    },
    {
        "name": "50 FPS",
        "value": 200000
    },
    {
        "name": "59.94 FPS",
        "value": 166833
    },
],

```

```

    {
      "name": "60 FPS",
      "value": 166667
    },
    {
      "name": "Follow Input",
      "value": 0
    }
  ],
  "profile": [
    {
      "name": "Baseline",
      "value": 0
    },
    {
      "name": "Main profile",
      "value": 1
    },
    {
      "name": "High profile",
      "value": 2
    }
  ],
  "hevc-profile": [
    {
      "name": "Main profile",
      "value": 0
    }
  ],
  "video-kbps": [
    {
      "name": "256 Kbps",
      "value": 256
    },
    {
      "name": "512 Kbps",
      "value": 512
    },
    {
      "name": "768 Kbps",
      "value": 768
    },
    {
      "name": "1 Mbps",
      "value": 1024
    },
    {
      "name": "1.5 Mbps",
      "value": 1536
    },
    {
      "name": "2 Mbps",
      "value": 2048
    },
    {
      "name": "3 Mbps",
      "value": 3072
    },
    {
      "name": "4 Mbps",
      "value": 4096
    },
    {
      "name": "5 Mbps",
      "value": 5120
    },
    {
      "name": "6 Mbps",
      "value": 6144
    },
    {
      "name": "8 Mbps",
      "value": 8192
    }
  ]

```

```
    },
    {
      "name": "10 Mbps",
      "value": 10240
    },
    {
      "name": "12 Mbps",
      "value": 12288
    },
    {
      "name": "16 Mbps",
      "value": 16384
    }
  ],
  "audio-kbps": [
    {
      "name": " 16 Kbps",
      "value": 16
    },
    {
      "name": " 32 Kbps",
      "value": 32
    },
    {
      "name": " 48 Kbps",
      "value": 48
    },
    {
      "name": " 64 Kbps",
      "value": 64
    },
    {
      "name": " 96 Kbps",
      "value": 96
    },
    {
      "name": "128 Kbps",
      "value": 128
    },
    {
      "name": "192 Kbps",
      "value": 192
    },
    {
      "name": "256 Kbps",
      "value": 256
    }
  ],
  "gop-sec": [
    {
      "name": " 1 sec",
      "value": 1
    },
    {
      "name": " 2 sec",
      "value": 2
    },
    {
      "name": " 5 sec",
      "value": 5
    },
    {
      "name": "10 sec",
      "value": 10
    },
    {
      "name": "30 sec",
      "value": 30
    },
    {
      "name": "60 sec",
      "value": 60
    }
  ]
}
```



```

    ],
    "video-range": [
      {
        "name": "Full range (0-255)",
        "value": 1
      },
      {
        "name": "Limited range (16-235)",
        "value": 0
      }
    ],
    "stat-sec": [
      {
        "name": " 1 sec",
        "value": 1
      },
      {
        "name": " 5 sec",
        "value": 5
      },
      {
        "name": "10 sec",
        "value": 10
      },
      {
        "name": "30 sec",
        "value": 30
      },
      {
        "name": "60 sec",
        "value": 60
      }
    ],
    "video-codec": [
      {
        "name": "H.264",
        "value": 0
      },
      {
        "name": "H.265 (HEVC)",
        "value": 1
      }
    ]
  ],
  "color-range": {
    "contrast": {
      "max": 200,
      "min": 50,
      "def": 100
    },
    "brightness": {
      "max": 100,
      "min": -100,
      "def": 0
    },
    "saturation": {
      "max": 200,
      "min": 0,
      "def": 100
    },
    "hue": {
      "max": 90,
      "min": -90,
      "def": 0
    }
  },
  "video-format": {
    "input-color-fmt": [
      {
        "name": "RGB",
        "value": 1
      },
      {

```

```

        "name": "YUV BT.601",
        "value": 2
    },
    {
        "name": "YUV BT.709",
        "value": 3
    },
    {
        "name": "YUV BT.2020",
        "value": 4
    }
],
"output-color-fmt": [
    {
        "name": "YUV BT.601",
        "value": 2
    },
    {
        "name": "YUV BT.709",
        "value": 3
    }
],
"quant-range": [
    {
        "name": "Full range (0-255)",
        "value": 1
    },
    {
        "name": "Limited range (16-235)",
        "value": 2
    }
],
"sat-range": [
    {
        "name": "Full range (0-255)",
        "value": 1
    },
    {
        "name": "Limited range (16-235)",
        "value": 2
    },
    {
        "name": "Extended GAMUT range (1-254)",
        "value": 3
    }
]
],
"living": {
    "max-bandwidth": 16384,
    "srt": {
        "latency": [
            {
                "name": "30 ms",
                "value": 30
            },
            {
                "name": "60 ms",
                "value": 60
            },
            {
                "name": "120 ms",
                "value": 120
            },
            {
                "name": "500 ms",
                "value": 500
            },
            {
                "name": "1000 ms",
                "value": 1000
            },
            {
                "name": "5000 ms",

```

```
        "value": 5000
      },
      {
        "name": "8000 ms",
        "value": 8000
      }
    ],
    "bandwidth": [
      {
        "name": "10%",
        "value": 10
      },
      {
        "name": "25%",
        "value": 25
      },
      {
        "name": "50%",
        "value": 50
      },
      {
        "name": "75%",
        "value": 75
      }
    ],
    "aes": [
      {
        "name": "Not Used",
        "value": 0
      },
      {
        "name": "AES-128",
        "value": 16
      },
      {
        "name": "AES-192",
        "value": 24
      },
      {
        "name": "AES-256",
        "value": 32
      }
    ]
  ]
}
}
```

## get-status

Use this interface to obtain the real-time running status of the device, including status of live stream, firmware update, etc.

### HTTP Request

```
GET http://ip/usapi?method=get-status
```

Parameter	Description
method	get-status

### Response Body

JSON structure is as follows:

```
{
  "result": 0, // returned status
  "cur-status": 65552, // device running status mask
  "last-rec-status": 0,
  "cur-time": "2021-01-11 ...", // device time
  "box-name": "", // device name
  "sysstat": {}, // system status
  "live-status": { // live stream status
    "live": []
  },
  "upgrade-status": {}, // firmware update status
  "living-test": {}, // streaming test status
  "check-upgrade": {}, // online update detecting status
  "conn-wifi": {}, // Wi-Fi connection status
  "input-signal": {}, // input signal
  "wifi": {}, // Wi-Fi connection
  "softap": {}, // AP connection
  "eth": {}, // Ethernet connection
  "mobile": {}, // mobile broadband connection
  "upgrade": {}, // new firmware information
  "channel-count": 2,
  "vumeters": [
    33,
    32
  ]
}
```

### Response Body

```
"result": 0
```

#### Device Running Status Mask

```
"cur-status": 65552
```

Masks vary depending on the various device running status. Refer to [Device Status Masks](#) to find specific description for each mask. The following condition is used to calculate the device running status.

1. Referring to the Device Status Masks, when a device is updating firmware, the mask is: `statusUpgrade = 0x4000`.
2. If `cur-status & statusUpgrade = statusUpgrade`, it indicates firmware update is in progress.

#### Input Signal Status

```
"input-signal": {
  "status": 0,
  "cx": 0,
  "cy": 0,
  "interlaced": 0,
```

```
"frame-rate": 0.00,  
"channel-valid": 0,  
"is-lpcm": 0,  
"bits-per-sample": 0,  
"sample-rate": 0  
}
```

#### Streaming Status

```
"live-status": {  
  [  
    {  
      "result": 0,  
      "run-ms": 0,  
      "cur-bps": 0,  
      "avg-bps": 0,  
      "net": 0,  
      "result2": 0,  
      "cur-bps2": 0,  
      "net2": 0,  
      "client-id": ""  
    }  
  ]  
}
```

#### Streaming Test Status

```
"living-test": {  
  "upload-bps": 0,  
  "percent": 0,  
  "result": 27,  
  "net": 0,  
  "client-id": ""  
}
```

#### Online Update Detecting Status

```
"check-upgrade": {  
  "result": 0,  
  "client-id": ""  
}
```

#### New Firmware Information

```
"upgrade": {  
  "ver": "",  
  "date": "",  
  "size-byte": 0,  
  "info": []  
}
```

#### Firmware Update Status

```
"upgrade-status": {  
  "step": 0,  
  "percent": 0,  
  "result": 27,  
  "client-id": "",  
  "mode": "none"  
}
```

#### Wi-Fi Connection

```
"wifi": {  
  "name": "MWL1",  
  "level": 0,  
  "ip": "192.168.8.249",  
}
```

```
"mask": "255.255.255.0",  
"router": "192.168.8.1",  
"dns": "192.168.8.1"  
}
```

#### **Ethernet Connection**

```
"eth": {  
  "ip": "10.10.107.212",  
  "mask": "255.255.0.0",  
  "router": "10.10.0.1",  
  "dns": "10.0.0.3"  
}
```

#### **Mobile Broadband Connection**

```
"mobile": {  
  "ip": "",  
  "mask": "",  
  "router": "",  
  "dns": ""  
}
```

# get-settings

Use this interface to obtain the device configurations.

## HTTP Request

```
GET http://ip/usapi?method=get-settings
```

Parameter	Description
method	get-settings

## Response Body

JSON structure is as follows:

```
{
  "result": 0, // returned status
  "name": "Ultra...", // device name
  "is-check-update": 1, // whether to auto-check latest firmware when connecting to Ethernet
  "audio-sync-offset": 0, // audio offset(ms)
  "softap": {}, // AP configurations
  "video-color": {}, // video information
  "volume": {}, // audio information
  "main-stream": {}, // main stream configurations
  "sub-stream": {}, // sub stream configurations
  "audio": {}, // audio configurations
  "eth": {}, // Ethernet information
  "stream-server": [...], // streaming session list
  "video-input-format": {} // input video format
  "video-output-format": {} // output video format
}
```

## Request Body

```
{
  "result": 0,
  "name": "Ultra Encode A304200908001",
  "passwd": 1,
  "is-check-update": 1,
  "audio-sync-offset": 0,
  "softap": {
    "is-softap": 1,
    "is-visible": 1,
    "softap-ssid": "00908001",
  },
  "video-color": {
    "contrast": 100,
    "brightness": 0,
    "saturation": 100,
    "hue": 0
  },
  "volume": {
    "is-mic": 1,
    "mic-gain": 0,
    "is-spi": 1,
    "spi-gain": 0,
    "is-phone": 1,
    "phone-gain": 0
  },
  "main-stream": {
    "is-auto": 0,
    "codec": 0,
    "cx": 1920,
    "cy": 1080,
    "duration": 166667,
    "kbps": 4096,
  }
}
```

```

    "gop": 1,
    "fourcc": 0,
    "profile": 2,
    "cbrstat": 60,
    "fullrange": 0
  },
  "sub-stream": {
    "cx": 1024,
    "cy": 768,
    "duration": 333333,
    "kbps": 1024,
    "gop": 1,
    "fourcc": 0,
    "profile": 2,
    "cbrstat": 60,
    "fullrange": 1
  },
  "audio": {
    "sample-rate": 48000,
    "channels": 2,
    "kbps": 128
  },
  "eth": {
    "is-dhcp": 1,
    "ip": "",
    "mask": "",
    "router": "",
    "dns": ""
  },
  "stream-server": [
    {
      "id": 0,
      "type": 0,
      "url": "192.168.1.123:345/live",
      "key": "aa",
      "is-auth": 0,
      "user": "",
      "passwd": "",
      "is-use": 0,
      "token": "",
      "net-mode": 1,
      "name": "192.168.1.123"
    }
  ]
  "video-input-format": {
    "is-color-fmt": 0,
    "color-fmt": 3,
    "is-quant-range": 0,
    "quant-range": 2
  },
  "video-output-format": {
    "is-color-fmt": 0,
    "color-fmt": 3,
    "is-quant-range": 0,
    "quant-range": 2,
    "is-sat-range": 0,
    "sat-range": 2
  }
}

```



# ping

Use this interface to check whether the device is accessible without log-in.

This function is used to ensure that the device has restarted after `firmware update` , `reset all settings` or `change IP address` .

## HTTP Request

```
GET http://ip/usapi?method=ping
```

Parameter	Description
method	ping

## Response Body

```
{  
  "result": 0,  
  "cur-status": 65552  
}
```

Item	Description
result	0 indicates the device is ready. Refer to <a href="#">API Status Codes</a> to find specific description for other values.
cur-status	Indicates the mask of current running status. Refer to <a href="#">Device Status Mask</a> to find specific description for each value.

# get-signal-info

Use this interface to obtain the information of input audio and video signals.

## HTTP Request

```
GET http://ip/usapi?method=get-signal-info
```

Parameter	Description
method	get-signal-info

## Response Body

```
{
  "result": 0
  "signal-info-types": [
    "video-info",
    "audio-info",
    "hdmi-info"
  ],
  "hdmi-info": {
    "mode": "hdmi",
    "vic": 0,
    "hdcpc": false,
    "it-content": false,
    "3d-format": false,
    "timing-h-total": 1650,
    "timing-h-active": 1280,
    "timing-h-frontporch": 110,
    "timing-h-syncwidth": 40,
    "timing-h-backporch": 220,
    "timing-f0v-syncwidth": 5,
    "timing-f0v-frontporch": 5,
    "timing-f0v-backproch": 20,
    "timing-f0v-active": 720,
    "timing-f0v-totalheight": 750
  },
  "audio-info": {
    "codec": "lpcm",
    "num-channels": 2,
    "sample-rate": 48000,
    "bit-count": 16
  },
  "video-info": {
    "codec": "uncompressed",
    "width": 1280,
    "height": 720,
    "scan": "progressive",
    "field-rate": 60.00,
    "color-depth": 8,
    "color-format": "bt.709",
    "aspect-ratio": "16:9",
    "sampling": "4:4:4",
    "quant-range": "limited",
    "sat-range": "limited",
    "frame-struct": "2d"
  }
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.



# get-report

Use this interface to check device information, status and configurations.

## HTTP Request

```
GET http://ip/usapi?method=get-report
```

Parameter	Description
method	get-report

## Response Body

```
{  
  <div class="report-summary">  
    ...  
    <h2>DEVICE</h2>  
    ...  
    <h2>STATUS</h2>  
    ...  
    <h2>SETTINGS</h2>  
    ...  
  </div>  
}
```

# export-report

Use this interface to export a .html file including device information, status and configurations.

## HTTP Request

```
GET http://ip/usapi?method=export-report&file-name=...
```

Parameter	Description
method	export-report
file-name	Path to which the exported file is saved

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## set-name

Use this interface to set device name.

### HTTP Request

```
GET http://ip/usapi?method=set-name&name=xxx
```

Parameter	Description
method	set-name
name	Device name ranging from 1 to 32 characters, which can consist of A to Z, a to z, 0 to 9, spaces . _ - ' [ ] ( ) , and cannot start or end with space.

### Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## set-first-over

Use this interface to initialize the encoder when accessing it for the first time, mainly to set the **device name**.

Obtain current device running status mask using [get-status](#).

```
{
  "cur-status": 65552           // device running status mask
  ...
}
```

[statusFirst\(0x01\)](#) indicates the device's first booting. If **cur-status & statusFirst = statusFirst**, it means the encoder is on the first run.

## HTTP Request

```
GET http://ip/usapi?method=set-first-over&name=xxx
```

Parameter	Description
method	set-first-over
name	Device name which contains 1. 1 to 32 characters 2. A to Z, a to z, 0 to 9, spaces and special characters <code>._-+'[]()</code> , and can not start or end with spaces.

## Response Body

```
{
  "result": 0
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# set-softap

Added in V1.3

Use this interface to modify AP configurations.

## HTTP Request

```
GET http://ip/usapi?method=set-softap&is-softap=0&is-visible=1&softap-ssid=xxx&softap-passwd=xxx
```

Parameter	Description
method	set-softap
is-softap	Reserved. The default value is 1.
is-visible	Reserved. The default value is 1.
softap-ssid	Reserved. The default value is the same as the product serial number.
softap-passwd	Indicates the SSID password in plain text. The string ranges from 1 to 32 characters which should contain A-Z, a-z, 0-9, spaces and special characters like . _ - + [ ] ( ). The password can not start or end with spaces.

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.



# set-ssdp

Added in V1.3

Use this interface to set whether to enable UPNP.

## HTTP Request

```
GET http://ip/usapi?method=set-ssdp&is-ssdp=1
```

Parameter	Description
method	set-ssdp
is-ssdp	Indicates whether to enable UPNP. 0: disable 1: enable

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# set-volume

added in V1.3

Use this interface to adjust the gain of input signal, headphones, and microphone.

You can obtain the value range of recording parameters using [get-info](#).

```
"audio-range": {  
  "hdm": [],      // input gain value  
  "mic": [],      // microphone gain value  
  "phone": [],    // headphones gain value  
}
```

## Response Body

```
GET http://ip/usapi?method=set-volume&is-mic=1&mic-gain=0&is-spi=1&spi-gain=0&is-phone=1&phone-gain=0
```

Parameter	Description
method	set-volume
is-mic	Indicates whether the microphone gain is enabled. 0: mute. 1: enable.
mic-gain	Indicates the microphone gain in dB. The default value is 0.
is-spi	Indicates whether the input gain is enabled. 0: mute. 1: enable.
spi-gain	Indicates the input gain in dB. The default value is 0.
is-phone	Indicates whether the headphone gain is enabled. 0: mute. 1: enable.
phone-gain	Indicates the headphone gain in dB. The default value is 0.

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# set-video-color

added in V1.3

Use this interface to adjust the brightness, contrast, hue and saturation of the image.

You can obtain the value range of recording parameters using [get-info](#).

```
"color-range": {  
  "contrast": [],      // Value range of contrast  
  "brightness": [],   // Value range of brightness  
  "saturation": [],   // Value range of saturation  
  "hue": []           // Value range of hue  
}
```

## HTTP Request

```
GET http://ip/usapi?method=set-video-color&contrast=100&brightness=0&saturation=100&hue=0
```

Parameter	Description
method	set-video-color
contrast	Indicates the contrast value. The default value is 100.
brightness	Indicates the brightness value. The default value is 0.
saturation	Indicates the saturation value. The default value is 100.
hue	Indicates the hue value. The default value is 0.

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# set-video-input-format

Added in V1.4

Use this interface to set color space and quantization for input video.

Obtain value range of video format parameters by calling [get-info](#).

```
"video-format": {  
  "input-color-fmt": [],      // Value range of color space  
  "quant-range": [],        // Value range of quantization  
}
```

## HTTP Request

```
GET http://ip/usapi?method=set-video-input-format&is-color-fmt=0&color-fmt=1&is-quant-range=0&quant-range=1
```

Parameter	Description
method	set-video-input-format
is-color-fmt	Whether to customize color space. The default value is 0, which indicates to auto-set color space, while 1 indicates a custom setting.
color-fmt	Indicates the color format parameter value. The value range is obtained by calling <a href="#">get-info</a> .
is-quant-range	Whether to customize quantization. The default value is 0, which indicates to auto-set quantization, while 1 indicates a custom setting.
quant-range	Indicates the quantization parameter value. The value range is obtained by calling <a href="#">get-info</a> .

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# set-video-output-format

Added in V1.4

Use this interface to set color space and quantization for output video.

Obtain value range of video format parameters by calling [get-info](#).

```
"video-format": {  
  "output-color-fmt": [],      // value range of color space  
  "quant-range": [],         // value range of quantization range  
  "sat-range": [],          // value range of saturation range  
}
```

## HTTP Request

```
GET http://ip/usapi?method=set-video-output-format&is-color-fmt=0&color-fmt=1&is-quant-range=0&quant-range=1&is-sat-range=0&sat-range=1
```

Parameter	Description
method	set-video-output-format
is-color-fmt	Whether to customize color space. The default value is 0, which indicates to auto-set color space, while 1 indicates a custom setting.
color-fmt	Indicates the color format parameter value.
is-quant-range	Whether to customize quantization. The default value is 0, which indicates to auto-set quantization, while 1 indicates a custom setting.
quant-range	Indicates the quantization parameter value.
is-sat-range	Whether to customize saturation. The default value is 0, which indicates to auto-set saturation, while 1 indicates a custom setting.
sat-range	Indicates the saturation range. The default value is 2.

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# set-sync-offset

added in V1.3

Use this interface to set audio offset in milliseconds. Set a positive value to delay the start of the audio track, or set a negative value to reduce the delay.

## HTTP Request

```
GET http://ip/usapi?method=set-sync-offset&audio-sync-offset=100
```

Parameter	Description
method	set-sync-offset
audio-sync-offset	Set audio offset, ranging from -200 to 200ms.

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## add-server

Use this interface to add streaming sessions, up to 16 tasks are supported to be added.

Different sessions with same configurations are not allowed, such as:

- Custom RTMP servers with the same URL and StreamKey
- Other RTMP servers(such as Twitch/YouTube/Facebook) with the same StreamKey
- RTSP sessions with the same port number
- SRT Caller sessions with the same IP address and port
- SRT Listenersessions with the same port
- HLS sessions with the same main stream name or sub stream name
- TS over UDP/RTP sessions with the same IP address and port

## HTTP Request

```
GET http://ip/usapi?method=add-server&type=xxx&url=xxx&key=xxx&...
```

Parameter	Description
method	add-server
type	Indicates a stream type. 0: RTMP 1: Twitch 2: YouTube 3: Facebook 100: RTSP 120: SRT Caller 121: SRT Listener 130: NDI HX 131: HLS 132: TS over UDP 133: TS over RTP
name	Indicates a session name, ranging from 1 to 32 characters, consisting of A to Z, a to z, 0 to 9, spaces ._-+'[()], and cannot start or end with space.
RTMP streaming settings	
stream-index	Indicates the stream type. 0: main stream 1: sub stream
url	Indicates streaming destination address.
key	Indicates stream key.
is-auth	Indicates authentication status. 0: authentication is not required 1: authentication is required. Both username and password are mandatory.
user	Indicates user name.
passwd	Indicates password.
token	Indicates Token.
event-data	Indicates streaming event.
net-mode	Indicates the network priority. 0: Mobile broadband first 1: Ethernet first 2: Wi-Fi first
RTSP streaming settings	
port	Indicates port number, the default port is 554.
max-connection	Indicates number of clients for each RTSP stream.
is-main	Indicates whether to enable the main stream. 0: disable the main stream. 1: enable the main stream.
main-stream-name	Indicates the main stream name, ranging from 1 to 32 characters, consisting of A to Z, a to z, 0 to 9, spaces, and special characters ._-+'[()]. The name should not start or end with spaces.
	Indicates whether to enable the sub stream.

is-sub	0: disable the sub stream. 1: enable the sub stream.
sub-stream-name	Indicates the sub stream name, ranging from 1 to 32 characters, consisting of A to Z, a to z, 0 to 9, spaces, and special characters ._-+'[](). The name should not start or end with spaces.
is-audio	Indicates whether to enable audio. 0: disable audio. 1: enable audio.
SRT Caller streaming settings	
stream-index	Indicates the stream type. 0: main stream 1: sub stream
url	Indicates destination address.
port	Indicates port number.
latency	Indicates delay time. You can obtain its value range by using <a href="#">get-info</a> .
bandwidth	Indicates the portion of the total bandwidth of a stream required for the exchange of SRT control and recovered packets. You can obtain its value range by using <a href="#">get-info</a> .
aes	Indicates the stream encryption algorithm to ensure the data security. 0: no encryption 16: AES-128 24: AES-192 32: AES-256
ase-word	Indicates passphrase.
stream-id	Optional
net-mode	Indicates network priority. 0: mobile broadband first 1: Ethernet first 2: Wi-Fi first
SRT Listener streaming settings	
stream-index	Indicates the stream type. 0: main stream 1: sub stream
port	Indicates service port number. The default port is 8000.
max-connection	Indicates max client number.
latency	Indicates delay time. You can obtain its value range by using <a href="#">get-info</a> .
bandwidth	the portion of the total bandwidth of a stream required for the exchange of SRT control and recovered packets. You can obtain its value range by using <a href="#">get-info</a> .
aes	Indicates the stream encryption algorithm to ensure the data security. 0: no encryption 16: AES-128 24: AES-192 32: AES-256
ase-word	Indicates passphrase.
NDI HX streaming settings	
source-name	By default, it is serial number.
group-name	Indicates the Group name which the source belongs to. By default, it is public.
enable-discovery	Whether to enable discovery server.
discovery-server	Indicates the IP address of discovery server.
enable-fail-over	Whether to enable fail over.
fail-over-ndi-name	Indicates the backup NDI channel name.
fail-over-ip-addr	Indicates the backup NDI channel IP address.
enable-web-control	Whether to allow to open the Web UI by clicking the gear icon in the NDI Studio Monitor application.
HLS streaming settings	
is-main	Whether to push streams over main stream. 0: disable 1: enable
main-stream-name	Indicates the main stream name, ranging from 1 to 32 characters, consisting of A to Z, a to z, 0 to 9, spaces, and special characters ._-+'[](). The name should not start or end with spaces.



is-sub	Whether to push streams over sub stream 0: disable 1: enable
sub-stream-name	Indicates the sub stream name, ranging from 1 to 32 characters, consisting of A to Z, a to z, 0 to 9, spaces, and special characters ._-+'[](). The name should not start or end with spaces.
TS over UDP/RTP streaming settings	
stream-index	Indicates the stream type. 0: main stream 1: sub stream
url	Indicates the destination address.
port	Indicates the stream port
net-mode	Indicates the network priority. 0: Mobile broadband first 1: Ethernet first 2: Wi-Fi first

## Response Body

```
{
  "result": 0
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## enable-server

Use this interface to turn on the stream session. By default, the newly added session is disabled.

Allow simultaneous sessions over the same streaming protocol, such as TS over RTP and TS over UDP, SRT Caller and SRT Listener.

- Up to 6 sessions of TS over RTP and TS over UDP simultaneously.
- Up to 6 sessions of SRT Caller and SRT Listener simultaneously, containing one SRT Listener session at most.
- Up to 6 sessions of RTMP and RTMPS simultaneously. It means that as long as the destination server allows, RTMP/RTMPS video platforms such as Twitch, Facebook, YouTube can be streamed to at the same time with your accounts.
- Stream only 1 session of HLS (push), RTSP, or NDI HX (NDI|HX2).

## HTTP Request

```
GET http://ip/usapi?method=enable-server&id=1&is-use=1
```

Parameter	Description
method	enable-server
id	Indicates the server ID, obtaining using <a href="#">get-settings</a> .
is-use	Indicates whether the server is enabled. 0: disable. 1: enabled.

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	Returned status. 0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## set-server

Use this interface to modify stream server.

### HTTP Request

```
GET http://ip/usapi?method=set-server&id=xxx&type=xxx&url=xxx&key=xxx&...
```

Parameter	Description
method	set-server
id	Indicates server ID, obtaining using <a href="#">get-settings..</a>
type	Indicates server type. 0: RTMP 1: Twitch 2: YouTube 3: Facebook 100: RTSP 120: SRT Caller 121: SRT Listener 130: NDI HX 131: HLS 132: TS over UDP 133: TS over RTP
name	Indicates server name, ranging from 1 to 32 characters, consisting of A to Z, a to z, 0 to 9, spaces, and special characters <code>._-+[]()</code> . The name should not start or end with spaces.
RTMP streaming settings	
stream-index	Indicates the stream type. 0: main stream 1: sub stream
url	Indicates streaming server address.
key	Indicates stream key.
is-auth	Indicates whether authentication is required. 0: authentication is not required. 1: authentication is required. Both username and password are mandatory.
user	Indicates user name.
passwd	Indicates password.
token	Indicates Token.
event-data	Indicates streaming event.
net-mode	Indicates the network priority. 0: Mobile network first 1: Ethernet first 2: Wi-Fi first
RTSP streaming settings	
port	Indicates port number, the default port is 554.
max-connection	Indicates number of clients for each RTSP stream.
is-main	Indicates whether to enable the main code stream. 0: disable the main stream. 1: enable the main stream.
main-stream-name	Indicates the main code stream name, ranging from 1 to 32 characters, consisting of A to Z, a to z, 0 to 9, spaces, and special characters <code>._-+[]()</code> . The name should not start or end with spaces.
is-sub	Indicates whether to enable the sub code stream. 0: disable the sub stream. 1: enable the sub stream.
sub-stream-name	Indicates the sub code stream name, ranging from 1 to 32 characters, consisting of A to Z, a to z, 0 to 9, spaces, and special characters <code>._-+[]()</code> . The name should not start or end with spaces.
is-audio	Indicates whether to enable audio. 0: disable audio. 1: enable audio.
SRT Caller	

streaming settings	
stream-index	Indicates the stream type. 0: main stream 1: sub stream
url	Indicates destination address.
port	Indicates port number.
latency	Indicates delay time. You can obtain its value range by using <a href="#">get-info</a> .
bandwidth	Indicates the portion of the total bandwidth of a stream required for the exchange of SRT control and recovered packets. You can obtain its value range by using <a href="#">get-info</a> .
aes	Indicates the stream encryption algorithm to ensure the data security. 0: no encryption 16: AES-128 24: AES-192 32: AES-256
ase-word	Indicates passphrase.
stream-id	Optional
net-mode	Indicates network priority. 0: mobile broadband first 1: Ethernet first 2: Wi-Fi first
SRT Listener streaming settings	
stream-index	Indicates the stream type. 0: main stream 1: sub stream
port	Indicates service port number. The default port is 8000.
max-connection	Indicates max client number.
latency	Indicates delay time. You can obtain its value range by using <a href="#">get-info</a> .
bandwidth	the portion of the total bandwidth of a stream required for the exchange of SRT control and recovered packets. You can obtain its value range by using <a href="#">get-info</a> .
aes	Indicates the stream encryption algorithm to ensure the data security. 0: no encryption 16: AES-128 24: AES-192 32: AES-256
ase-word	Indicates passphrase.
NDI HX streaming settings	
source-name	By default, it is serial number.
group-name	Indicates the Group name which the source belongs to. By default, it is public.
enable-discovery	Whether to enable discovery server.
discovery-server	Indicates the IP address of discovery server.
enable-fail-over	Whether to enable fail over.
fail-over-ndi-name	Indicates the backup NDI channel name.
fail-over-ip-addr	Indicates the backup NDI channel IP address.
enable-web-contro	Whether to allow to open the Web UI by clicking the gear icon in the NDI Studio Monitor application.
HLS streaming settings	
is-main	Whether to push streams over main stream. 0: disable 1: enable
main-stream-name	Indicates the main stream name, ranging from 1 to 32 characters, consisting of A to Z, a to z, 0 to 9, spaces, and special characters <code>._-+[]()</code> . The name should not start or end with spaces.
is-sub	Whether to push streams over sub stream 0: disable 1: enable
sub-stream-name	Indicates the sub stream name, ranging from 1 to 32 characters, consisting of A to Z, a to z, 0 to 9, spaces, and special characters <code>._-+[]()</code> . The name should not start or end with spaces.
TS over UDP/RTP streaming settings	

stream-index	Indicates the stream type. 0: main stream 1: sub stream
url	Indicates the destination address.
port	Indicates the stream port
net-mode	Indicates the network priority. 0: Mobile broadband first 1: Ethernet first 2: Wi-Fi first

## Response Body

```
{
  "result": 0
}
```

Item	Description
result	Returned status. 0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# del-server

Use this interface to delete stream sessions.

## HTTP Request

```
GET http://ip/usapi?method=del-server&id=1
```

Parameter	Description
method	del-server
id	Indicates the server ID, obtaining using <a href="#">get-settings</a> .

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	Returned status. 0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## start-test-server

Use this interface to start a stream test. Only one stream can be tested each time.

Call the [stop-test-server](#) to cancel the stream test manually if needed. The test will stop automatically at 20s.

### HTTP Request

```
GET http://ip/usapi?method=start-test-server&type=xxx&url=xxx&key=xxx&...
```

Parameter	Description
method	start-test-server
type	Indicates the session type. 0: RTMP 1: Twitch 2: YouTube 3: Facebook 120: SRT Caller
name	Indicates server name, ranging from 1 to 32 characters, consisting of A to Z, a to z, 0 to 9, spaces, and special characters <code>._-+[]()</code> . The name should not start or end with spaces.
RTMP streaming settings	
stream-index	Indicates the stream type. 0: main stream 1: sub stream
url	Indicates destination address.
key	Indicates the stream key.
is-auth	Indicates whether an authentication is required. 0: authentication is not required 1: authentication is required, and both username and password are mandatory.
user	Indicates username.
passwd	Indicates password.
token	Indicates token.
event-data	Indicates a stream session.
net-mode	Indicates the network priority. 0: Mobile broadband first 1: Ethernet first 2: Wi-Fi first
SRT Caller streaming settings	
stream-index	Indicates the stream type. 0: main stream 1: sub stream
url	Indicates destination address.
port	Indicates stream port.
latency	Indicates delay time. You can obtain its value range by using <a href="#">get-info</a> .
bandwidth	Indicates the portion of the total bandwidth of a stream required for the exchange of SRT control and recovered packets. You can obtain its value range by using <a href="#">get-info</a> .
aes	Indicates the stream encryption algorithm to ensure the data security. 0: no encryption 16: AES-128 24: AES-192 32: AES-256
ase-word	Indicates passphrase.
stream-id	Optional
net-mode	Indicates the prime network for streaming. 0: Mobile Broadband first 1: Ethernet first 2: Wi-Fi first

## Response Body

```
{
  "result": 0
}
```

Item	Description
result	Returned status. 0: stream test is started. 1: repeated request. -9: system is busy. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## Test Status

Obtain **living-test** by calling [get-status](#).

```
"living-test": {
  "result": 27,
  "upload-bps": 0,
  "percent": 0,
  "net": 0,
  "client-id": ""
}
```

Item	Description
result	Returned status. 27: the device is in initial status. 5 indicates that request is canceled. 2: stream test is in progress. 0: the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.
percent	Indicates the percentage of stream test.
upload-bps	Indicates the upload speed.
net	Indicates the network type.
client-id	Reserved

Call the [clear-test-server](#) to set the device to the initial status (retlnit=27) after the test.



# stop-test-server

Use this interface to cancel stream test after starting a test by calling [start-test-server](#).

## HTTP Request

```
GET http://ip/usapi?method=stop-test-server
```

Parameter	Description
method	stop-test-server

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	Returned status. 0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## clear-test-server

Use this interface to reset the device to the initial state (reInIt=27) after the device conducts stream test using [start-test-server](#).

### HTTP Request

```
GET http://ip/usapi?method=clear-test-server
```

Parameter	Description
method	clear-test-server

### Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	Returned status. 0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# get-ndi-sources

Use this interface to obtain other NDI senders in the current LAN when NDI|HX service is enabled.

## HTTP Request

```
GET http://ip/usapi?method=get-ndi-sources
```

Parameter	Description
method	get-ndi-sources

## Response Body

```
{  
  "result": 0  
  "ndi-sources": [  
    {  
      "ndi-name": "PRO CONVERT (#14 (B403190104002))",  
      "ip-addr": "10.10.13.247:5962"  
    }  
  ]  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## set-video

Use this interface to configure parameters of the main and sub streams, including:

- resolution
- frame interval
- video codec: choose H264 or HEVC for the main stream. The sub stream uses H264.
- profile
- video bit rate
- keyframe interval
- quantization range
- bit rate stats duration

Obtain parameters value range using [get-info](#).

```
{
  "codec-cap": {
    "resolutions": [],           // Value range of resolution
    "durations": [],            // Value range of frame interval
    "video-codec": [],          // Value range of video codec
    "profile": [],              // H264 profile
    "hevc-profile": [],         // HEVC profile
    "video-kbps": [],           // Value range of video bit rate
    "gop-sec": [],              // Value range of keyframe interval
    "video-range": [],          // Value range of quantization range
    "stat-sec": []              // Value range of bit rate stats duration
  }
}
```

## HTTP Request

```
GET http://ip/usapi?method=set-video&stream=0&is-auto=0&cx=1280&cy=720&duration=333333&kbps=1024&gop=1&fourcc=0&profile=2&cbrstat=60&fullrange=1&codec=0
```

Parameter	Description
method	set-video
stream	Indicates the stream type. 0: the main stream 1: the sub stream
is-auto	Indicates whether the stream format follows that of input signal. For a sub stream, this parameter can only be set to 0. 0: custom the format for your session. 1: follow input.
cx	Indicates width of resolution dimensions in pixels.
cy	Indicates height of resolution dimensions.
duration	Indicates frame interval.
kbps	Indicates bit rate.
gop	Indicates keyframe interval.
fourcc	Reserved. The default value is 0.
codec	Indicates video codec.
profile	Indicates encoding profile.
cbrstat	Indicates bit rate stats duration.
fullrange	Indicates quantization range.

## Response Body

```
{
  "result": 0
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## set-audio

Use this interface to configure audio encoding parameters.

Obtain value range of audio bit rate using [get-info](#).

```
{
  "codec-cap": {
    "audio-kbps": [],          // Value range of audio bit rate
  }
}
```

### HTTP Request

```
GET http://ip/usapi?method=set-audio&kbps=48
```

Parameter	Description
method	set-audio
kbps	Indicates audio bit rate.

### Response Body

```
{
  "result": 0
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# reboot

Use this interface to reboot and reconnect to your device.

The reboot process may take a few minutes. You can use [ping](#) to determine whether the restart is finished.

## HTTP Request

```
GET http://ip/usapi?method=reboot
```

Parameter	Description
method	reboot

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## reset-all-settings

Use this interface to reset all settings back to default.

The reset process may take a few minutes, and all configuration data will be lost. After resetting, the device will restart. You can use the [ping](#) interface to check the device restart status.

### HTTP Request

```
GET http://ip/usapi?method=reset-all-settings
```

Parameter	Description
method	reset-all-settings

### Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.



# login

Use this interface to log in. The cookie will carry the session ID after you log in successfully. For example, Cookie:  
sid=e0f6b33dd2b575eff40733b3778beaab.

## HTTP Request

```
GET http://ip/usapi?method=login&id=xxx&pass=xxx
```

Parameter	Description
method	login
id	Indicates user name.
pass	Indicates MD5 encrypted password.

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0: the request was accepted successfully. 36: the username or password is incorrect. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# logout

Use this interface to log out and return to the **SIGN IN** page.

## HTTP Request

```
GET http://ip/usapi?/usapi?method=logout
```

Parameter	Description
method	logout

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## get-users

Use this interface to list all users with administrative rights.

### HTTP Request

```
GET http://ip/usapi?method=get-users
```

Parameter	Description
method	get-users

### Response Body

```
{
  "result": 0,
  "users": [
    {
      "id": "Admin",
      "type": 1
    },
    {
      "id": "Test",
      "type": 2
    }
  ]
}
```

Item	Description
result	Returned status. 0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.
users	Indicates list of users. id indicates user name. type indicates whether the user is an administrator(1) or a general user(2).

## add-user

Use this interface to add general users with administrative rights.

### HTTP Request

```
GET http://ip/usapi?method=add-user&id=xxx&pass=xxx
```

Parameter	Description
method	add-user
id	Indicates the user name.
pass	Indicates MD5 encrypted password.

### Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	Returned status. 0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## del-user

Use this interface to delete general users with administrative rights.

### HTTP Request

```
GET http://ip/usapi?method=del-user&id=xxx
```

Parameter	Description
method	del-user
id	Indicates the user name.

### Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	Returned status. 0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# ch-password

Use this interface to modify the password after logging in.

## HTTP Request

```
GET http://ip/usapi?method=ch-password&pass=xxx&new-pass=xxx
```

Parameter	Description
method	ch-password
pass	Indicates MD5 encrypted old password.
new-pass	Indicates MD5 encrypted new password.

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	Returned status. 0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## set-password

Use this interface to reset user password without entering the old password. Only admin user is allowed to perform the operation.

### HTTP Request

```
GET http://ip/usapi?method=set-password&id=xxx&pass=xxx
```

Parameter	Description
method	set-password
id	Indicates the user name.
pass	Indicates MD5 encrypted new password.

### Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	Returned status. 0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## set-eth

Use this interface to configure Ethernet network.

### HTTP Request

```
GET http://ip/usapi?method=set-eth&is-dhcp=1&ip=10.10.107.212&mask=255.255.0.0&router=10.10.0.1&dns=10.0.0.3
```

Parameter	Description
method	set-eth
is-dhcp	Indicates whether to obtain an IP address from the DHCP server. 0: set IP address manually. 1: obtain an IP address from the DHCP server dynamically.
ip	Indicates Ethernet IP address.
mask	Indicates the subnet mask.
router	Indicates the gateway.
dns	Indicates DNS server.

### Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.



# scan-wifi-results

Use this interface to obtain available Wi-Fi networks.

## HTTP Request

```
GET http://ip/usapi?method=scan-wifi-results
```

Parameter	Description
method	scan-wifi-results

## Response Body

```
{
  "result": 0
  "user-items": [
  ],
  "scan-items": [
  ]
}
```

### Wi-Fi Network Connected

```
"user-items": [
  {
    "name": "",
    "freq": 5785,
    "level": -35,
    "is-auto": 0,
    "is-use": 0
  }
]
```

### Wi-Fi Network Not Connected

```
"scan-items": [
  {
    "name": "",
    "freq": 5785,
    "level": -35,
    "is-auto": 0,
    "is-use": 0
  }
]
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## connect-wifi-first

Use this interface to connect your device to a Wi-Fi network for the first time.

### HTTP Request

```
GET http://ip/usapi?method=connect-wifi-first&name=magewell&passwd=1111111&secu=2&is-auto=1
```

Parameter	Description
method	connect-wifi-first
name	Indicates name of Wi-Fi network.
passwd	Indicates password of Wi-Fi network.
secu	Indicates encryption method of Wi-Fi network: 0 - unencrypted; 1 - wep; 2 - WPA/PSK; 3 - WPA2/PSK.
is-auto	Indicates whether to auto-connect to this Wi-Fi network.

### Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## connect-wifi

Use this interface to connect to a saved Wi-Fi network.

### HTTP Request

```
GET http://ip/usapi?method=connect-wifi&name=magewellis-auto=1
```

Parameter	Description
method	connect-wifi
name	Indicates name of Wi-Fi network.
is-auto	Indicates whether to auto-connect to this Wi-Fi network.

### Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# cancel-connect-wifi

Use this interface to cancel current Wi-Fi connection.

## HTTP Request

```
GET http://ip/usapi?method=cancel-connect-wifi&name=magewell
```

Parameter	Description
method	cancel-connect-wifi
name	Wi-Fi network name

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## disconnect-wifi

Use this interface to disconnect from a Wi-Fi Network.

### HTTP Request

```
GET http://ip/usapi?method=disconnect-wifi
```

Parameter	Description
method	disconnect-wifi

### Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# forget-wifi

Use this interface to forget a Wi-Fi network and delete information about the hot spot from your device.

## HTTP Request

```
GET http://ip/usapi?method=forget-wifi&name=magewell
```

Parameter	Description
method	forget-wifi
name	Indicate a Wi-Fi network name

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## set-connect-wifi-auto

Use this interface to set whether to allow your device to connect automatically to a Wi-Fi network.

### HTTP Request

```
GET http://ip/usapi?method=set-connect-wifi-auto&name=magewell&is-auto=1
```

Parameter	Description
method	set-connect-wifi-auto
name	Indicate Wi-Fi network name
is-auto	whether to allow the device to connect automatically to a specific Wi-Fi

### Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## clear-connect-wifi

Use this interface to reset device to initial state (retInit=27) when it fails to configure Wi-Fi using [connect-wifi-first](#).

### HTTP Request

```
GET http://ip/usapi?method=clear-connect-wifi
```

Parameter	Description
method	clear-connect-wifi

### Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.



# open-softap

added in V1.3

Use this interface to turn on the AP mode.

## HTTP Request

```
GET http://ip/usapi?method=open-softap
```

Parameter	Description
method	open-softap

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# close-softap

Added in V1.3

Use this interface to turn off AP mode.

## HTTP Request

```
GET http://ip/usapi?method=close-softap
```

Parameter	Description
method	close-softap

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# update

There are two ways to perform the firmware update:

- Manual update: upload the specified firmware file using [upload-update-file](#) before update.
- Online update: use [online-update-check](#) to check if there is a new version available.

In order to perform the firmware update:

- Firmware checking must not be in progress.

You can obtain current device running status mask using [get-status](#).

```
"cur-status": 65552 // device running status mask
```

The output of `cur-status` & [Device Status Mask](#) is as follows:

Device status	Condition
Firmware update is not in progress	cur-status & statusCheckUpgrade != statusCheckUpgrade

## HTTP Request

```
GET http://ip/usapi?method=update&mode=xxx
```

Parameter	Description
method	update
mode	Firmware update type. upload: manual update. online: check and update automatically.

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	Returned status. 0 indicates the encoder begins to update. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## Update Status

Obtain `upgrade-status` using [get-status](#).

```
"upgrade-status": {  
  "result": 27,  
  "step": 0,  
  "percent": 0,  
  "mode": "none",  
  "client-id": ""  
}
```

Item	Description
result	Indicates returned update status. 27: initial status. 2: updating. 0: the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.
step	Indicates update steps.
percent	Indicates the percentage of the update step.
mode	Indicates firmware update type. none: initial status. upload: manual update

	online: online update
client-id	Reserved.

Call [clear-upgrade](#) to reset the status to 27 (Initial status) after update failed.

# upload-update-file

Use this interface to upload the .mwf firmware file to your device.

## HTTP Request

```
POST http://ip/usapi?method=upload-update-file
```

Parameter	Description
method	upload-update-file

## Response Body

```
{  
  "status": 0,  
  "up-to-date": true,  
  "version": "1.1.72",  
  "size": 11890776  
}
```

Item	Description
status	Indicates returned status. 0 indicates that the operation is performed successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.
up-to-date	Indicates whether the current firmware is up to date. If yes, it is true; otherwise, it is false.
version	Indicates the uploaded firmware version number.
size	Indicates the uploaded file size in bytes.

Call [update](#) to update the unit after a successful upload.

# cancel-download

Use this interface to cancel the firmware downloading when the firmware is updating online using the [update](#) interface.

## HTTP Request

```
GET http://ip/usapi?method=cancel-download
```

Parameter	Description
method	cancel-download

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	Indicates returned status. 0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## online-update-check

Use this interface to check for new firmware versions.

- In order to conduct the firmware checking, an update can not be in progress.

You can obtain current device running status mask using [get-status](#):

```
"cur-status": 65552 // device running status mask
```

The output of `cur-status` & [Device Status Mask](#) is as follows:

Device status	Condition
Firmware update is not in progress	cur-status & statusUpgrade != statusUpgrade

## HTTP Request

```
GET http://ip/usapi?method=online-update-check
```

Parameter	Description
method	online-update-check

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

## Detection Status

Obtain `check-upgrade` and `upgrade` using [get-status](#).

```
"check-upgrade": {  
  "result": 0,  
  "client-id": ""  
},  
"upgrade": {  
  "ver": "1.2.123",  
  "date": "2012-1-1 00:00:00",  
  "size-byte": 12004784,  
  "info": [  
    {  
      "version": "1.2.123",  
      "changelog": "## Develop version 1.2. ## Develop version2."  
    }  
  ]  
}
```

Item	Description
result	Indicates returned status. 27: initial status 2: checking 0: the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.
ver	Indicates new firmware version.
date	Indicates release date of the new firmware.
size-byte	Indicates file size of new firmware(B).
info	Indicates release note.

client-id	Reserved.
-----------	-----------

Call [clear-check-update](#) to reset the status to Initial status(27) after detection failed.



## clear-upgrade

Use this interface to reset the device to the initial state (retlnit=27) when the device fails to update manually or automatically using [update](#).

### HTTP Request

```
GET http://ip/usapi?method=clear-upgrade
```

Parameter	Description
method	clear-upgrade

### Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	Indicates returned status. 0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# clear-check-update

Use this interface to reset the device to the initial state (retlnit=27) when new firmware check fails or succeeds using the [online-update-check](#).

## HTTP Request

```
GET http://ip/usapi?method=clear-check-update
```

Parameter	Description
method	clear-check-update

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	Indicates returned status. 0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.

# set-enable-check-update

Added in V1.4

Use this interface to set whether to allow the device to automatically and regularly check for new firmware versions.

## HTTP Request

```
GET http://ip/usapi?method=set-enable-check-update&is-check-update=1
```

Parameter	Description
method	set-enable-check-update
is-check-update	0: disable auto-check 1: enable auto-check

## Response Body

```
{  
  "result": 0  
}
```

Item	Description
result	0 indicates that the request was accepted successfully. Refer to <a href="#">API Status Codes</a> to find specific description for other values.