

ANT-2601 API Guide

v2.02 - 2019.01.08

HDMI & DVI H.265 + H264 4KP30 Video Decoder



Decoder API- 2019.01.08 (v2.02)

Protocol port: 80

About header

Each HTTP header required authorization information, such as "**Authorization:Basic YWRtaW46YWRtaW4=**", this header means authorization information that includes user name and password.

```
Accept:application/json, text/javascript, */*
Accept-Encoding:gzip, deflate
Accept-Language:zh-CN,zh;q=0.9
Authorization:Basic YWRtaW46YWRtaW4=
Connection:keep-alive
Host:192.168.1.160
Referer:http://192.168.1.160/
User-Agent:Mozilla/5.0 (iPhone; CPU iPhone OS 10_3 like Mac OS X) AppleWebKit/602.1.50 (KHTML, like Gecko) CriOS/56.0.2924
X-Requested-With:XMLHttpRequest
```

"**YWRtaW46YWRtaW4=**" is Base64 encoded value, base64 decoding softwares will be able to detect the value:
admin:admin

Adding "authentication" title required when requesting HTTP access, here is value for reference regards Java jointing authorization header

```
String admin= "admin"
String pwd = "admin"
String login = admin + ":" + pwd; //user name and passwordthe jointed login value is admin:admin
byte[] encode = android.util.Base64.encode(login.getBytes(), android.util.Base64.DEFAULT); //Base64 in encrypting
String enc = new String(encode); // convert byte to stringthe converted enc value is YWRtaW46YWRtaW4=
String auth = "Basic " + enc //
```

Different HTTP request lib has different way to add header, pasting the illustration regards how to add okhttp lib in Java

```
public static OkHttpClient genericClient() {
    OkHttpClient httpClient = new OkHttpClient.Builder()
        .addInterceptor(new Interceptor() {
            @Override
            public Response intercept(Chain chain) throws IOException {
                Request request = chain.request()
                    .newBuilder()
                    .addHeader("Content-Type", "application/x-www-form-urlencoded; charset=UTF-8")
                    .addHeader("Accept-Encoding", "gzip, deflate")
                    .addHeader("Connection", "keep-alive")
                    .addHeader("Accept", "*/*")
                    .addHeader("Authorization", "Basic YWRtaW46YWRtaW4=")
                    .build();
                return chain.proceed(request);
            }
        })
        .build();

    return httpClient;
}
```

/get_status Functions

get paying states

value

No

request

POST|GET

illustration

http://192.168.0.170/open_get_status

return: (JSON)

successfully returned:

```
{
  "status": 0,
  "result": {
    "output_screen_resolution": 0,
    "split_screen_count": 0,
    "video_stream_urls": [
      "http://192.168.0.88:80/hdmi",
      "http://192.168.0.88:80/hdmi"
    ]
  }
}
```

/set_net

Function

Set up eth and wlan0 information, uploaded Json data includes eth0 and wlan0 object, following functions filled in eth0 and WLAN0, (WLAN0 will be working when using WiFi)

Functions

Functions	Type	Value	Necessary	Description	Note
dhcp	string	0/1	Y	dhcp switch	DHCP enable, statics IP will be disabled
ip	string	Free IP	N	Ethernet ip	For statics IP
mask	string	255.255.255.0	N	Ethernet mask	For statics IP

gateway	string	Router gateway	N	Ethernet default gateway	For statics IP
dns	string	Usable DNS	N	DNS server address	For statics IP

Request

POST|GET

illustration:

```
http://192.168.0.170/set_net?request={"eth0":{"dhcp":"0","ip":"192.168.0.178","mask":"255.255.255.0","gateway":"192.168.0.1"},
"dns":"114.114.114.114"},"wlan0":{"dhcp":"0","ip":"192.168.0.178","mask":"255.255.255.0","gateway":"192.168.0.1"},
"dns":"114.114.114.114"}}
```

return:

```
{
  "message": "succeeded",
  "status": 0
}
```

/get_net

Function

Getting network adapter information

parameter

No

request

POST|GET

illustration:

http://192.168.0.170/get_net

Return result (JSON)

```
{
  "netcfg": {
    "ethernet": {
      "dhcp": false,
      "dns1": "223.5.5.5",
      "gateway": "192.168.1.1",
      "ip": "192.168.0.178",
      "netmask": "255.255.255.0"
    },
    "wifi": {
      "dhcp": false,
      "encryption": 0
    }
  },
  "status": true
}
```

/get_codec

Function

Getting the detailed configuration information

parameter

No

###request

POST|GET

illustration:

http://192.168.0.170/get_codec

return result(JSON)

```
{
  "data": {
    "hdmi_mode": "1280x720p-60",
    "hdmi_modes": [
      "auto",
      "3840x2160p-30",
      "3840x2160p-25",
      "3840x2160p-24",
      "1920x1080p-60",
      "1920x1080p-30",
      "1920x1080p-25",
      "1920x1080p-24",
      "1280x720p-60",
      "1280x720p-50",
      "720x576p-50",
      "720x480p-60"
    ],
    "addr_way": "0",
    "input_urls": "http://192.168.0.217:80/hdmi",
    "output_mode_selected": 0,
    "play_files": "es.sdp",
    "split_mode": "0"
  },
  "status": 0
}
```

/set_codec

Configuring decoder

Parameters

Name	Type	value	Necessary	Description	Note
request	string		Y	1*json	loads streaming URL

What can be added behind of request:

Parameters	Type	Value	Necessary	Description	Note
split_mode	string	"0" "1" "2" "3"	Y	how many screen need to split	current 1 only, so "0"
addr_way	string	"0""1"	Y	input path	"0"-URL, "1" file
input_urls	string	"" or available address	Y	Input URL	4 byte limited array, set up one URL
play_files	string	""or available file name	Y	File name	4 byte limited array, set up one file name

request

POST|GET

illustration:

http://192.168.0.170/set_codec?request={"split_mode":"0","addr_way":"0","input_urls":["http://192.168.0.158:80/hdmi"],"play_files":["es.sdp"]}

return result(JSON)

```
{
  "message": "succeeded",
  "status": 0
}
```

/set_player_state

Getting playing states

Parameters

Parameters	Type	value	Necessary	Description	Note
player_state	string	play,pause,stop,restart	Y	load the request, such as "pause"	-

Request

POST|GET

illustration:

```
http://192.168.0.170/set_player_state?player_state=pause
```

return result

```
{"message": "succeeded", "status": 0}
```

/get_player_state

Getting playing states

request

POST|GET

illustration:

```
http://192.168.0.170/get_player_state
```

return result

```
{"player_state": "pause", "message": "succeeded ", "status": 0}
```

result note

Three player_state: play,pause,stop.

/upload_sdp_data

updating/ uploading SDP file

parameters

parameter	Type	Value	Necessary	Description	Note
file_name	string		Y	SDP name	-
file_data	string		Y	SDP data	string

request

POST

illustration:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta content="text/html; charset=utf-8" http-equiv="Content-Type">
  <title>sdp </title>
  <meta name="viewport" content="width=device-width,initial-scale=1"/>
  <script src="Js/jquery-1.4.1.min.js" type="text/javascript"></script>
  <script type="text/javascript">
    var hostip = window.location.host;
    function add_sdp_data() {
      var file_name = $("file_name").val;
      var file_data = $("file_data").val;
      $.ajax({
        type: "POST",
        url: "http://" + hostip + "/upload_sdp_data",
        dataType: "json",
        data: {
          "file_name": file_name,
          "file_data": file_data
        },
        cache: false,
        success: function (data) {
          if (data.status == 0) {
            alert('succeeded ');
          } else {
            alert('fault ');
          }
        }
      });
    }
  </script>
</head>
<body>
file name <input type="text" name="file_name" id="file_name"></br>
file data <input type="text" name="file_data" id="file_data"></br>
<button type="button" onclick="add_sdp_data()">set</button>
</body>
</html>
```

return value(JSON)

```
{
  "message": "succeeded",
  "status": 0
}
```

/get_transcode

function

Getting transcoder information

parameter

--

POST|GET

illustration:

http://192.168.0.170/get_transcode

return result(JSON)

```
{
  "data": {
    "transcoder_open": "1"
    "format": "flv",
    "url_in": "rtsp://admin:hwsx123456@192.168.0.64:554/h264/ch1/main/av_stream",
    "url_out": "rtmp://192.168.0.21:1935/live/decoder",
  },
  "status": 0
}
```

/set_transcode

configuring transcoder

parameter

Parameter	Type	Value	Necessary	Description	Note
request	string		Y	1*json	loads streaming URL

###What can be added behind of request

parameters	type	value	necessary	description
transcoder_open	string	"0" or "1"	Y	"0"-disable,"1"- enable
url_in	string	"" or Streaming URL	Y	The video sources
url_out	string	"" or the output URL	Y	The output address
format	string	"flv" or "mpegts" or "mp4"	Y	transcoding format

request

POST|GET

illustration:

http://192.168.0.170/set_transcode?request=

```
{"transcoder_open":"1","url_in":"rtsp://admin:hwsx123456@192.168.0.64:554/h264/ch1/main/av_stream","url_out":"rtmp://192.168.0.21:1935/live/decoder","format":"flv"}
```

return result (JSON)

```
{
  "message": "succeeded",
  "status": 0
}
```

/reboot

###function###

Reboot system

parameter

reboot

request

POST|GET

illustration:

<http://192.168.0.170/reboot>

return result(JSON)

#

/file_up_load

function

Upload files

header

content-type need to be multipart/form-data

parameter

Parameter	Type	Value	Necessary	Description	Note
-----------	------	-------	-----------	-------------	------

request

POST|GET

illustration:

http://192.168.0.170/file_up_load

return result(JSON)

#

/set_hdmi_mode

function

Configuring HDMI output resolution

http://192.168.0.170/set_hdmi_mode?hdmi_mode=1920x1080p-60

Note: parameters might be different for different instance