HDTV0404N301
API Command Set

Version: V1.0.2

Default RS232 Settings

Parameters	Value
Baud Rate	115200 bps
Data bits	8 bits
Parity	None
Stop bits	1 bit
Flow control	None

Telnet Connection

Before you intend to control the device through telnet API, you shall establish telnet connection between this device and your computer.
The form of the command for telnet connection is below:
telnet ip (port)
ip: The device's IP address.
port: The device's port number, this is non-required on some Telnet control tools or platforms. Default setting is 23.
For example, if the device's IP address is 192.168.11.143, the command for telnet connection shall be *telnet 192.168.11.143*
Note: Default IP setting for this device is DHCP. To obtain its IP address, please send command *GET IPADDR<CR><LF>* in the following command list to the device using a RS232 serial tool.

Command Explanation

Take Command SET SW in <CR><LF> as an example:
1. [SET SW] denotes command key words, case insensitive.
2. [in] denotes parameters, case insensitive; incorrect parameters number will not be recognized.
3. <CR><LF> denotes a carriage return or a line feed; all commands must be ended up with a carriage return or a line feed.

Command List

No.	Description	Command	Example
		Video Mode	
1	Set video mode	**Command:** SET VIDMODE prm <CR><LF> **Return:** VIDMODE prm<CR><LF> **Parameter:** prm = {Matrix, VideoWall, MultiView} **Description:** Set video output mode.	**Command:** SET VIDMODE Matrix<CR><LF> **Return:** VIDMODE Matrix<CR><LF> **Description:** Set video output mode.
2	Get video mode	**Command:** GET VIDMODE <CR><LF> **Return:** VIDMODE prm<CR><LF> **Parameter:** prm = {Matrix, VideoWall, MultiView} **Description:** Get video output mode.	**Command:** GET VIDMODE <CR><LF> **Return:** VIDMODE Matrix<CR><LF> **Description:** Get video output mode.
		Multiple View	
3	Set Video output display layout	**Command:** SET VIDOUT_MODE [prm] <CR><LF> **Return:** VIDOUT_MODE prm<CR><LF> **Parameter:** prm = {0, 1, 2, 4, 5} // 0: Original // 1: Dual-view // 2: Pip // 4: Master // 5: Quad	**Command:** SET VIDOUT_MODE 0<CR><LF> **Return:** VIDOUT_MODE 0<CR><LF> **Description:** Set Video output display layout as original.

No.	Description	Command	Example
		//optional parameter, switch from 0 to 5 without parameter **Description:** Set Video output display layout.	
4	Get Video output display layout	**Command:** GET VIDOUT_MODE <CR><LF> **Return:** VIDOUT_MODE prm<CR><LF> **Parameter:** prm = {0, 1, 2, 4, 5} // 0: Original // 1: Dual-view // 2: Pip // 4: Master // 5: Quad **Description:** Get video output resolution mode.	**Command:** GET VIDOUT_MODE<CR><LF> **Return:** VIDOUT_MODE 0<CR><LF> **Description:** Get video output resolution mode.
5	Set video source in dual layout mode	**Command:** SET VIDOUT_DUAL_SRC prm1 prm2 <CR><LF> **Return:** VIDOUT_DUAL_SRC prm1 prm2<CR><LF> **Parameter:** prm1 = {in1, in2…in4}; //left channel prm2 = {in1, in2…in4};//right channel **Description:** Set video source in dual layout mode. You must be ensured in dual mode for this operation.	**Command:** SET VIDOUT_DUAL_SRC in1 in2<CR><LF> **Return:** VIDOUT_DUAL_SRC in1 in2<CR><LF> **Description:** Set video source in dual layout mode.

No.	Description	Command	Example
6	Get video source in dual layout mode	**Command:** GET VIDOUT_DUAL_SRC <CR><LF> **Return:** VIDOUT_DUAL_SRC prm1 prm2<CR><LF> **Parameter:** prm1 = {in1, in2…in4}; //left channel prm2 = {in1, in2…in4};//right channel **Description:** Get video source in dual layout mode.	**Command:** GET VIDOUT_DUAL_SRC<CR><LF> **Return:** VIDOUT_DUAL_SRC in1 in2<CR><LF> **Description:** Get video source in dual layout mode.
7	Set video source in PIP layout mode	**Command:** SET VIDOUT_PIP_SRC prm prm1<CR><LF> **Return:** VIDOUT_PIP_SRC prm prm1<CR><LF> **Parameter:** prm = {in1, in2…in4} // big channel prm 1= {in1, in2…in4}//small channel **Description:** Set video source in dual layout mode. You must be ensured in pip mode for this operation.	**Command:** SET VIDOUT_PIP_SRC in1 in2<CR><LF> **Return:** VIDOUT_PIP_SRC in1 in2<CR><LF> **Description:** Set video source in dual layout mode.

No.	Description	Command	Example
10	Get video source in PIP layout mode	**Command:** GET VIDOUT_PIP_SRC <CR><LF> **Return:** VIDOUT_PIP_SRC prm prm1<CR><LF> **Parameter:** prm = {in1, in2…in4} // big channel prm1= {in1, in2…in4}//small channel **Description:** Get video source in PIP layout mode.	**Command:** GET VIDOUT_PIP_SRC<CR><LF> **Return:** VIDOUT_PIP_SRC in1 in2<CR><LF> **Description:** Get video source in PIP layout mode.
11	Set video source in QUAD layout mode	**Command:** SET VIDOUT_QUAD_SRC prm prm1 prm2 prm3<CR><LF> **Return:** VIDOUT_QUAD_SRC prm prm1 prm2 prm3<CR><LF> **Parameter:** prm = {in1, in2…in4} // top of the left channel prm1 = {in1, in2…in4} //top of the right channel prm2 = {in1, in2…in4} //bottom of the left channel prm3 = {in1, in2…in4} //bottom of the right channel **Description:** Set video source in QUAD layout mode. You must be ensured in QUAD mode for this operation.	**Command:** SET VIDOUT_QUAD_SRC in1 in2 in3 in4<CR><LF> **Return:** VIDOUT_QUAD_SRC in1 in2 in3 in4<CR><LF> **Description:** Set video source in QUAD layout mode.

No.	Description	Command	Example
12	Get video source in QUAD layout mode	**Command:** GET VIDOUT_QUAD_SRC <CR><LF> **Return:** VIDOUT_QUAD_SRC prm prm1 prm2 prm3<CR><LF> **Parameter:** prm = {in1, in2…in4} // top of the left channel prm1 = {in1, in2…in4} //top of the right channel prm2 = {in1, in2…in4} //bottom of the left channel prm3 = {in1, in2…in4} //bottom of the right channel **Description:** Get video source in QUAD layout mode.	**Command:** GET VIDOUT_QUAD_SRC <CR><LF> **Return:** VIDOUT_QUAD_SRC in1 in2 in3 in4<CR><LF> **Description:** Get video source in QUAD layout mode.
13	Set video source in MASTER layout mode	**Command:** SET VIDOUT_MASTER_SRC prm prm1 prm2 prm3<CR><LF> **Return:** VIDOUT_MASTER_SRC prm prm1 prm2 prm3<CR><LF> **Parameter:** prm = {in1, in2…in4} // left channel prm1 = {in1, in2…in4} //top of the right channel prm2 = {in1, in2…in4} //middle of the left channel prm3 = {in1, in2…in4} //bottom of the right channel **Description:** Set video source in MASTER layout mode. You must be ensured in MASTER mode for this operation.	**Command:** SET VIDOUT_MASTER_SRC in1 in2 in3 in4<CR><LF> **Return:** VIDOUT_MASTER_SRC in1 in2 in3 in4<CR><LF> **Description:** Set video source in MASTER layout mode.

No.	Description	Command	Example
14	Get video source in MASTER layout mode	**Command:** GET VIDOUT_MASTER_SRC <CR><LF> **Return:** VIDOUT_MASTER_SRC prm prm1 prm2 prm3<CR><LF> **Parameter:** prm = {in1, in2…in4} // left channel prm1 = {in1,in2…in4} //top of the right channel prm2 = {in1,in2…in4} //middle of the left channel prm3 = {in1,in2…in4} //bottom of the right channel **Description:** Get video source in MASTER layout mode.	**Command:** GET VIDOUT_MASTER_SRC <CR><LF> **Return:** VIDOUT_MASTER_SRC in1 in2 in3 in4<CR><LF> **Description:** Get video source in MASTER layout mode.
15	Set pip small window size	**Command:** SET VIDOUT_PIP_SIZE prm <CR><LF> **Return:** VIDOUT_PIP_SIZE prm <CR><LF> **Parameter:** prm = {0, 1, 2} // 0: 1/4 // 1: 1/9 // 2: 1/16 **Description:** Set pip small window size	**Command:** SET VIDOUT_PIP_SIZE 0 <CR><LF> **Return:** VIDOUT_PIP_SIZE 0 <CR><LF> **Description:** Set pip small window size

No.	Description	Command	Example
16	Get pip small window size	**Command:** GET VIDOUT_PIP_SIZE <CR><LF> **Return:** VIDOUT_PIP_SIZE prm <CR><LF> **Parameter:** prm = {0, 1, 2} // 0: 1/4 // 1: 1/9 // 2: 1/16 **Description:** Get pip small window size.	**Command:** GET VIDOUT_PIP_SIZE<CR><LF> **Return:** VIDOUT_PIP_SIZE 0 <CR><LF> **Description:** Get pip small window size.
17	Set pip small window position	**Command:** SET VIDOUT_PIP_POS prm<CR><LF> **Return:** VIDOUT_PIP_POS prm <CR><LF> **Parameter:** prm = {0,1, 2, 3} // 0: top of the left // 1: top of the right // 2: bottom of the left // 3: bottom of the right **Description:** Set pip small window position.	**Command:** SET VIDOUT_PIP_POS 0 <CR><LF> **Return:** VIDOUT_PIP_POS 0 <CR><LF> **Description:** Set pip small window position.
18	Get pip small window position	**Command:** GET VIDOUT_PIP_POS<CR><LF> **Return:** VIDOUT_PIP_POS prm <CR><LF> **Parameter:** prm = {0,1， 2, 3} // 0: top of the left // 1: top of the right	**Command:** GET VIDOUT_PIP_POS <CR><LF> **Return:** VIDOUT_PIP_POS 0 <CR><LF> **Description:** Get pip small window position.

No.	Description	Command	Example
		// 2: bottom of the left // 3: bottom of the right **Description:** Get pip small window position.	
19	Set input video stretch status	**Command:** SET VIDIN_STRETCH prm prm1 <CR><LF> **Return:** VIDIN_STRETCH prm prm1 <CR><LF> **Parameter:** prm = {in1, in2…in4} prm1 = {origin, :full} **Description:** Set input video stretch status	**Command:** SET VIDIN_STRETCH in1 origin <CR><LF> **Return:** VIDIN_STRETCH in1 origin<CR><LF> **Description:** Set input video stretch status.
20	Get input video stretch status	**Command:** GET VIDIN_STRETCH prm <CR><LF> **Return:** VIDIN_STRETCH prm prm1 <CR><LF> **Parameter:** prm = {in1, in2…in4} prm1 = {origin, :full} **Description:** Get input video stretch status.	**Command:** GET VIDIN_STRETCH in1<CR><LF> **Return:** VIDIN_STRETCH in1 origin<CR><LF> **Description:** Get input video stretch status.
21	Set audio output window	**Command:** SET AUDOUT_WND prm <CR><LF> **Return:** AUDOUT_WND prm<CR><LF> **Parameter:** prm = {in1, in2…in4}, is optional parameter. Reset to current window without	**Command:** SET AUDOUT_WND in1 <CR><LF> **Return:** AUDOUT_WND in1<CR><LF> **Description:** Set audio output window.

No.	Description	Command	Example
		parameter **Description:** Set audio output window	
22	Get audio output window	**Command:** GET AUDOUT_WND <CR><LF> **Return:** AUDOUT_WND prm<CR><LF> **Parameter:** prm = {in1, in2…in4} **Description:** Get audio output window.	**Command:** GET AUDOUT_WND <CR><LF> **Return:** AUDOUT_WND in1<CR><LF> **Description:** Get audio output window.
Video Wall			
1	Set video wall and select input source for video wall	**Command:** SET VIDWALL *in prm1 prm2 prm3 prm4*<CR><LF> **Return:** VIDWALL in *prm1 prm2 prm3 prm4*<CR><LF> **Parameter:** *prm1* = {out1, out2…out4}; prm2 = {out1, out2…out4}; prm3 = {out1, out2…out4}; prm4 = {out1, out2…out4}; *in* = {in1, in2…in4}; **Description:** VIDWALL is short for Video wall. Set video wall and select input source for video wall.	**Command:** SET VIDWALL *in1 out2 out1 out3 out4*<CR><LF> **Return:** VIDWALL *in1 out2 out1 out3 out4*<CR><LF> **Description:** Set video wall and select input source for video wall.

No.	Description	Command	Example
2	Get Video Wall settings and the input source selected	**Command:** GET VIDWALL<CR><LF> **Return:** VIDWALL in *prm1 prm2 prm3 prm4*<CR><LF> **Parameter:** *prm1* = {out1, out2…out4}; prm2 = {out1, out2…out4}; prm3 = {out1, out2…out4}; prm4 = {out1, out2…out4}; *in* = {in1, in2…in4}; **Description:** VIDWALL is short for Video wall. Get Video Wall settings and the input source selected.	**Command:** GET VIDWALL<CR><LF> **Return:** VIDWALL *in1 out2 out1 out3 out4*<CR><LF> **Description:** Get Video Wall settings and the input source selected.
3	Set Video Wall bezel correction	**Command:** SET VIDWALLBEZEL VW OW VH OH <CR><LF> **Return:** VIDWALLBEZEL VW OW VH OH <CR><LF> **Parameter:** VW = {0-10000} OW = {0-10000} VH = {0-10000} OH = {0-10000} **Description:** Set Video Wall bezel correction.	**Command:** SET VIDWALLBEZEL 2 2 2 2 <CR><LF> **Return:** VIDWALLBEZEL 2 2 2 2<CR><LF> **Description:** Set Video Wall bezel correction.

No.	Description	Command	Example
4	Get Video Wall bezel correction	**Command:** GET VIDWALLBEZEL <CR><LF> **Return:** VIDWALLBEZEL VW OW VH OH <CR><LF> **Parameter:** VW = {0-10000} OW = {0-10000} VH = {0-10000} OH = {0-10000} **Description:** Get Video Wall bezel correction.	**Command:** GET VIDWALLBEZEL <CR><LF> **Return:** VIDWALLBEZEL 2 2 2 2<CR><LF> **Description:** Get Video Wall bezel correction.
5	Set 180° rotation to enable/disable	**Command:** SET VIDWALL_ROTATION prm1 prm2 <CR><LF> **Return:** VIDWALL_ROTATION prm1 prm2<CR><LF> **Parameter:** prm1 = {out1, out2…out4}; prm2 = {disable, enable} **Description:** Set 180° rotation to enable/disable.	**Command:** SET VIDWALL_ROTATION out2 disable <CR><LF> **Return:** VIDWALL_ROTATION out2 disable<CR><LF> **Description:** Set 180°rotation of output2 to disable.
6	Get whether 180° rotation is set to enable or disable	**Command:** GET VIDWALL_ROTATION prm1 <CR><LF> **Return:** VIDWALL_ROTATION prm1 prm2<CR><LF> **Parameter:** prm1 = {out1, out2…out4}; prm2 = {disable, enable} **Description:** Get whether 180°rotation is set to enable or disable.	**Command:** GET VIDWALL_ROTATION out2 <CR><LF> **Return:** VIDWALL_ROTATION out2 disable<CR><LF> **Description:** 180°rotation of output 2 is set to disable.

No.	Description	Command	Example
		Video Switch	
1	Switch Input to Output	**Command:** SET SW *in out*<CR><LF> **Return:** SW *in out*<CR><LF> **Parameter:** *in* = {in1, in2…in4}; *out* = {out1, out2…out4}; **Description:** in -in0 means power down output SW is short for Switch Switch one input source to one output sink.	**Command:** SET SW *in1 out2*<CR><LF> **Return:** SW *in1 out2*<CR><LF> **Description:** Switch input 1 to hdmi output 2.
2	Switch one input for all outputs	**Command:** SET SW *in all*<CR><LF> **Return:** SW *in all* <CR><LF> **Parameter:** *in* = {in1, in2…in4}; *all* = {all}; **Description:** in - in0 means power down output SW is short for Switch Switch one input source for all output sinks.	**Command:** SET SW *in1 all* <CR><LF> **Return:** SW *in1 all*<CR><LF> **Description:** Switch input1 for all output sinks.

No.	Description	Command	Example
3	Get which input is mapping to the indicate Output	**Command:** GET MP *out*<CR><LF> **Return:** Mp *in out*<CR><LF> **Parameter:** *in* = {in1, in2…in4}; *out* = {out1, out2…out4}; **Description:** in - in0 means power down output MP is short for mapping Get which input is mapping to the indicate Output.	**Command:** GET MP *out1*<CR><LF> **Return:** MP *in2 out1*<CR><LF> **Description:** Get which input is mapping to output 1, the result is input 2.
4	Get current corresponding relationships of all inputs and all outputs	**Command:** GET MP *all*<CR><LF> **Return:** MP *in out*<CR><LF> MP *in out*<CR><LF> ..... **Parameter:** *in* = {in1, in2…in4}; *out* = {out1, out2…out4}; *all* = {all}; **Description:** in - in0 means power down output MP is short for mapping Get current corresponding relationships of all inputs and all outputs.	**Command:** GET MP *all* <CR><LF> **Return:** MP *in1 out1*<CR>... MP *in2 out2*<CR><LF> …... **Description:** Get current corresponding relationships of all inputs and all outputs.

No.	Description	Command	Example
		Audio	
1	Set Audio Mute for specific Audio Output	**Command:** SET AUDIO_MUTE *out prm*<CR><LF> **Return:** AUDIO_MUTE *out prm*<CR><LF> **Parameter:** *out* = {zone1, zone2}; *prm* = {on, off}; **Description:** Set Audio Mute for specific Audio Output.	**Command:** SET AUDIO_MUTE *zone1 on*<CR><LF> **Return:** AUDIO_MUTE *zone1 on*<CR><LF> **Description:** Set zone1 mute to on.
2	Get Audio Mute Status of specific Audio Output	**Command:** GET AUDIO_MUTE *out*<CR><LF> **Return:** AUDIO_MUTE out<CR><LF> **Parameter:** out = {zone1, zone2}; prm = {on, off}; **Description:** Get Audio Mute Status of specific Audio Output.	**Command:** GET AUDIO_MUTE *zone1*<CR><LF> **Return:** AUDIO_MUTE *zone1* on<CR><LF> **Description:** Zone 1 is set to mute.

No.	Description	Command	Example
		CEC Control	
1	Set CEC POWER ON/OFF	**Command:** SET CEC_PWR *out prm*<CR><LF> **Return:** CEC_PWR *out prm*<CR><LF> **Parameter:** *prm* = {on, off} *out* = {out1, out2…out4, all}; **Description:** Set sink to power on or off.	**Command:** SET CEC_PWR *out1 on*<CR><LF> **Return:** CEC_PWR *out1 on*<CR><LF> **Description:** Set sink output 1 to power on.
2	Set CEC AUTO POWER ON/OFF	**Command:** SET AUTOCEC_FN *out prm*<CR><LF> **Return:** AUTOCEC_FN *out prm*<CR><LF> **Parameter:** *prm* = {on, off} *out* = {out1, out2…out4, all}; **Description:** Set sink auto power Function to ON or OFF.	**Command:** SET AUTOCEC_FN *out1 on*<CR><LF> **Return:** AUTOCEC_FN *out1 on*<CR><LF> **Description:** Set sink output 1 auto power function to ON.
3	Get CEC AUTO POWER ON/OFF Status	**Command:** GET AUTOCEC_FN *out*<CR><LF> **Return:** AUTOCEC_FN *out prm*<CR><LF> **Parameter:** *prm* = {on, off} *out* = {out1, out2…out4, all}; **Description:** Get Sink auto power Function ON or OFF Status.	**Command:** GET AUTOCEC_FN *out1*<CR><LF> **Return:** AUTOCEC_FN *out1on* **Description:** Get Sink auto power status, and the status is ON.

No.	Description	Command	Example
4	Set CEC POWER Delay Time	**Command:** SET AUTOCEC_D *out prm*<CR><LF> **Return:** AUTOCEC_D *out prm*<CR><LF> **Parameter:** *out* = {out1, out2…out4, all}; *prm* = {1~30}// according to the actual time counter,1 means 1 minute, 2 means 2 minutes，Default wait time is 2 minutes, Max wait time is 30 minutes. **Description:** AUTOCEC_D is short for CEC auto Power Delay Timing.	**Command:** SET AUTOCEC_D *out1 2*<CR><LF> **Return:** AUTOCEC_D *out1 2*<CR><LF> **Description:** when no active signal to hdmi1, 2 minutes later, the unit will auto power off.
5	Get CEC POWER Delay Time Status	**Command:** GET AUTOCEC_D *out*<CR><LF> **Return:** AUTOCEC_D *out prm*<CR><LF> **Parameter:** *out* = {out1, out2…out4, all}; *prm* = {1~30}// according to the actual time counter,1 means 1 minute, 2 means 2 minutes，Default wait time is 2 minutes, Max wait time is 30 minutes. **Description:** AUTOCEC_D is short for CEC auto Power Delay Timing.	**Command:** GET AUTOCEC_D *out1*<CR><LF> **Return:** AUTOCEC_D *out1 2*<CR><LF> **Description:** Get hdmi1 auto power delay time, the result is 2 minutes.

No.	Description	Command	Example
		HDCP	
1	Set Input HDCP support to ON/OFF	**Command:** SET HDCP_S *in prm*<CR><LF> **Return:** HDCP_S *in prm*<CR><LF> **Parameter:** *prm* = {on, off} *in = {in1, in2…in4, all}* **Description:** HDCP_S will control source hdcp support to on or off.	**Command:** SET HDCP_S *in1 on*<CR><LF> **Return:** HDCP_S *in1 on*<CR><LF> **Description:** Set hdmi input 1 hdcp support to on.
2	Get Input HDCP support ON/OFF Status	**Command:** GET HDCP_S *in* <CR><LF> **Return:** HDCP_S *in prm*<CR><LF> **Parameter:** *prm* = {on, off} *in = {in1, in2…in4, all}* **Description:** HDCP_S is short for HDCP support.	**Command:** GET HDCP_S in1<CR><LF> **Return:** HDCP_S *in1 on*<CR><LF> **Description:** Get hdmi1 hdcp support on or off status, and the result is on.

No.	Description	Command	Example
		EDID	
1	Set Input EDID	**Command:** SET EDID *in prm*<CR><LF> **Return:** EDID *in prm*<CR><LF> **Parameter:** *in = {in1, in2…in4, all}* *prm = {1 ~16}* 1: Copy form hdmi output 1 2: Copy form hdmi output 2 ... 4: Copy form hdmi output 4 5~10: Reserved 11: Fixed 4K60 2.0CH PCM Audio with HDR; 12: Fixed 4K60 2.0CH PCM Audio with SDR; 13: Fixed 4K30 2.0CH PCM Audio with HDR; 14: Fixed 4K30 2.0CH PCM Audio with SDR; 15: Fixed 1080p@60Hz 2.0CH PCM Audio with HDR; 16: Fixed 1080p@60Hz 2.0CH PCM Audio with SDR; **Description:** Set Input EDID.	**Command:** SET EDID *in1 12*<CR><LF> **Return:** EDID *in1 12*<CR><LF> **Description:** Set in1's EDID to Fixed 4K60 2.0CH PCM Audio with SDR.

No.	Description	Command	Example
2	Get All Input EDID status	**Command:** GET EDID *all*<CR><LF> **Return:** EDID *in prm*<CR> EDID *in prm*<CR> EDID *in prm*<CR><LF> ..... **Parameter:** in = {in1, in2...in8}; prm = {1 ~16} 1: Copy form hdmi output 1 2: Copy form hdmi output 2 ... 4: Copy form hdmi output 4 5~10: Reserved 11: Fixed 4K60 2.0CH PCM Audio with HDR; 12: Fixed 4K60 2.0CH PCM Audio with SDR; 13: Fixed 4K30 2.0CH PCM Audio with HDR; 14: Fixed 4K30 2.0CH PCM Audio with SDR; 15: Fixed 1080p@60Hz 2.0CH PCM Audio with HDR; 16: Fixed 1080p@60Hz 2.0CH PCM Audio with SDR; **Description:** Get all input EDID Status.	**Command:** GET EDID *all*<CR><LF> **Return:** EDID in1 *01*<CR> EDID in2 *02*<CR> EDID in3 *03*<CR><LF> …… **Description:** Get all input EDID Status.

No.	Description	Command	Example
3	Get one input EDID Status	**Command:** GET EDID *in*<CR><LF> **Return:** EDID *in prm*<CR><LF> **Parameter:** in = {in1, in2...in4}; prm = {1 ~16} 1: Copy form hdmi output 1 2: Copy form hdmi output 2 ... 4: Copy form hdmi output 4 5~10: Reserved 11: Fixed 4K60 2.0CH PCM Audio with HDR; 12: Fixed 4K60 2.0CH PCM Audio with SDR; 13: Fixed 4K30 2.0CH PCM Audio with HDR; 14: Fixed 4K30 2.0CH PCM Audio with SDR; 15: Fixed 1080p@60Hz 2.0CH PCM Audio with HDR; 16: Fixed 1080p@60Hz 2.0CH PCM Audio with SDR; **Description:** Get one input EDID Status.	**Command:** GET EDID *in1*<CR><LF> **Return:** EDID *hdmiin1 13*<CR><LF> **Description:** Get in1 edid status, and the status is Fixed 4K30 2.0CH PCM Audio with HDR.
4	Write EDID to one input	**Command:** SET EDID_W *in prm1 prm2*<CR><LF> **Return:** EDID_W *in prm1 prm3*<CR><LF> **Parameter:** *in = {in1, in2...in4, all};* *prm1 = {block0, block1};* *prm2* = one block of 256 bytes edid ascii data with spaces (hex data need conversion into ASCII code) *prm3* = {ok, error}; error:check	**Command:** SET EDID_W *in1 block0 XX…XX*<CR><LF> **Return:** EDID_W *in1 block0 ok*<CR><LF> **Description:** Write EDID content to input.

No.	Description	Command	Example
		sum error **Description:** Write EDID content to input.	
5	Read EDID of one output	**Command:** GET EDID_R *out*<CR><LF> **Return:** EDID_R *out prm1 prm2*<CR><LF> **Parameter:** *out* = {out1, out2...out4}; *prm1* = {block0, block1}; *prm2* = {one block of 256 bytes edid ascii data with no spaces (hex data need conversion into ASCII code), error, unconnect}; **Description:** Read EDID content form output.	**Command:** GET EDID_R *out1*<CR><LF> **Return:** EDID_R *out1 block0 XX…XX*<CR><LF> EDID_R *out1 block1 XX…XX*<CR><LF> **Description:** EDID_R *out1 block0 XX…XX*<CR><LF> --- Read EDID ok or EDID_R *out1 error*<CR><LF> --- Check Sum Error or EDID_R *out1 unconnect*<CR><LF> --- Sink unconnect
System Info			
1	Factory reset	**Command:** RESET<CR><LF> **Return:** RESET<CR><LF> **Description:** Factory reset.	**Command:** RESET<CR><LF> **Return:** RESET<CR><LF> **Description:** Factory reset all boards.
2	System reboot	**Command:** REBOOT<CR><LF> **Return:** REBOOT<CR><LF>	**Command:** REBOOT<CR><LF> **Return:** REBOOT<CR><LF>

No.	Description	Command	Example
		Description: System reboot.	**Description:** System reboot.
3	Set IR System Code	**Command:** Set IR_SC *prm* <CR><LF> **Return:** IR_SC *prm*<CR><LF> **Parameter:** *prm* = {*all, mode1, mode2*}; *mode1 = 0x00* *mode2 = 0x4e* **Description:** Set IR System Code.	**Command:** Set IR_SC *mode1*<CR><LF> **Return:** IR_SC *mode1*<CR><LF> **Description:** Set IR System code to mode 1.
4	Get IR System Code	**Command:** Get IR_SC <CR><LF> **Return:** IR_SC *prm*<CR><LF> **Parameter:** *prm* = {*all, mode1, mode2*}; *mode1 = 0x00* *mode2 = 0x4e* **Description:** Get IR System Code.	**Command:** Get IR_SC <CR><LF> **Return:** IR_SC *mode1*<CR><LF> **Description:** Get IR System code ，IR System code is mode 1.
5	Get the API list	**Command:** help<CR><LF> **Return:** xxxx **Description:** Get the API list.	**Command:** help<CR><LF> **Return:** xxxx **Description:** Get the API list

No.	Description	Command	Example
6	GET IP address	**Command:** GET IPADDR<CR><LF> **Return:** IPADDR xxx.xxx.xxx.xxx MASK xxx.xxx.xxx.xxx GATEWAY xxx.xxx.xxx.xxx<CR><LF> **Description:** GET IP address.	**Command:** GET IPADDR<CR><LF> **Return:** IPADDR 192.168.1.2 MASK 255.255.255.0 GATEWAY 192.168.2.1<CR><LF>
7	SET IP address	**Command:** SET IPADDR xxx.xxx.xxx.xxx MASK xxx.xxx.xxx.xxx GATEWAY xxx.xxx.xxx.xxx<CR><LF> **Return:** IPADDR xxx.xxx.xxx.xxx MASK xxx.xxx.xxx.xxx GATEWAY xxx.xxx.xxx.xxx<CR><LF> **Description:** SET IP address.	**Command:** SET IPADDR 192.168.1.2 MASK 255.255.255.0 GATEWAY 192.168.2.1<CR><LF> **Return:** IPADDR 192.168.1.2 MASK 255.255.255.0 GATEWAY 192.168.2.1<CR><LF>
8	GET Netconfig mode	**Command:** GET NETCFG MODE<CR><LF> **Return:** NETCFG MODE prm<CR><LF> **Parameter:** prm = {DHCP, STATIC}; **Description:** GET Netconfig mode.	**Command:** GET NETCFG MODE<CR><LF> **Return:** NETCFG MODE DHCP<CR><LF>

No.	Description	Command	Example
9	SET Netconfig mode	**Command:** SET NETCFG MODE prm<CR><LF> **Return:** NETCFG MODE prm<CR><LF> **Parameter:** prm = {DHCP, STATIC}; **Description:** SET Netconfig mode.	**Command:** SET NETCFG MODE DHCP<CR><LF> **Return:** NETCFG MODE DHCP<CR><LF>
10	Set the device to Standby mode	**Command:** STANDBY<CR><LF> **Return:** STANDBY!<CR><LF>	**Command:** STANDBY<CR><LF> **Return:** STANDBY!<CR><LF>
11	Wake the device	**Command:** WAKE<CR><LF> **Return:** WAKE!<CR><LF>	**Command:** WAKE<CR><LF> **Return:** WAKE!<CR><LF>
12	Get Standby	**Command:** GET STANDBY<CR><LF> **Return:** STANDBY!<CR><LF> or WAKE!<CR><LF>	**Command:** GET STANDBY<CR><LF> **Return:** STANDBY!<CR><LF> or WAKE!<CR><LF>

No.	Description	Command	Example
13	Get the connection status of the video input	**Command:** GET VIDIN_CONNECT *in*<CR><LF> **Return:** VIDIN_CONNECT *in prm*<CR><LF> *in = {in1, in2...in4};* *prm = {Disconnected, Connected}* **Description:** Get the connection status of the video input.	**Command:** GET VIDIN_CONNECT in1<CR><LF> **Return:** VIDIN_CONNECT in1 Disconnected<CR><LF>
14	Get the signal status of the video input	**Command:** GET VIDIN_SIG *in*<CR><LF> **Return:** VIDIN_SIG *in prm*<CR><LF> *in* = {in1, in2...in4}; *prm* = {no, valid} **Description:** Get the signal status of the video input.	**Command:** GET VIDIN_SIG in1<CR><LF> **Return:** VIDIN_SIG in1 valid<CR><LF>

No.	Description	Command	Example
15	Get input video format information	**Command:** GET VIDIN_FORMAT *in*<CR><LF> **Return:** VIDIN_FORMAT i*n prm*<CR><LF> in = {in1, in2...in4}; prm = { } **Description:** Get the resolution of the video input prm = {<horizontal>x<vertical>,<rate>; <HDR info>;<ColorSpace>,<DeepColor>} • horizontal = An integer value representing the horizontal. • vertical = An integer value representing the vertical. May have an additional qualifier such as 'i' or 'p'. • rate = An integer value representing the refresh rate. • HDR info = none hdr/ static hdr/ dynamic hdr • Color space = RGB / Ycbcr 444 /Ycbcr 422/Ycbcr 420 • DeepColor = 8 bit/10 bit /12 bit/ 16 bit	**Command:** GET VIDIN_FORMAT *in1*<CR><LF> **Return:** VIDIN_FORMAT *in1 3840*x2160,60;none hdr;rgb;8bit<CR><LF>

No.	Description	Command	Example
16	Get hdcp version of input video	**Command:** GET VIDIN_HDCP *in*<CR><LF> **Return:** VIDIN_HDCP *in prm*<CR><LF> *in = {in1, in2...in4};* *prm = {no hdcp, hdcp1.4, hdcp2.2}* **Description:** Get hdcp version of input video.	**Command:** GET VIDIN_HDCP *in1*<CR><LF> **Return:** VIDIN_HDCP *in1 no hdcp*<CR><LF>
17	Get the connection status of the video output	**Command:** GET VIDOUT_CONNECT *out*<CR><LF> **Return:** VIDOUT_CONNECT *in prm*<CR><LF> *out = {out1, out2...out4}*; *prm = {Disconnected, Connected}* **Description:** Get the connection status of the video output.	**Command:** GET VIDOUT_CONNECT out1<CR><LF> **Return:** VIDOUT_CONNECT out1 Disconnected<CR><LF>
18	Get the signal status of the video output	**Command:** GET VIDOUT_SIG *out*<CR><LF> **Return:** VIDOUT_SIG *out prm*<CR><LF> *out* = {out1, out2…out4}; *prm* = {no, valid} **Description:** Get the signal status of the video output.	**Command:** GET VIDOUT_SIG out1<CR><LF> **Return:** VIDOUT_SIG *out1 valid*<CR><LF>

No.	Description	Command	Example
19	Get the video format information of output	**Command:** GET VIDOUT_FORMAT *out*<CR><LF> **Return:** VIDOUT_FORMAT out *prm*<CR><LF> *out = {out1, out2…out4};* *prm = { }* **Description:** Get the resolution of the video input prm = {<horizontal>x<vertical>,<rate>; <HDR info>;<ColorSpace>,<DeepColor>} • horizontal = An integer value representing the horizontal. • vertical = An integer value representing the vertical. May have an additional qualifier such as 'i' or 'p'. • rate = An integer value representing the refresh rate. • HDR info = none hdr/ static hdr/ dynamic hdr • Color space = RGB / Ycbcr 444 /Ycbcr 422/Ycbcr 420 • DeepColor = 8 bit/10 bit /12 bit/ 16 bit	**Command:** GET VIDOUT_FORMAT *out1*<CR><LF> **Return:** VIDOUT_FORMAT *out1* *3840*x2160,60;none hdr;rgb;8bit<CR><LF>

No.	Description	Command	Example
20	Get hdcp version of output video	**Command:** GET VIDOUT_HDCP *out*<CR><LF> **Return:** VIDOUT_HDCP *out prm*<CR><LF> *out = {out1, out2…out4};* *prm = {no hdcp, hdcp1.4, hdcp2.2}* **Description:** Get hdcp version of output video.	**Command:** GET VIDOUT_HDCP *out1*<CR><LF> **Return:** VIDOUT_HDCP *out1 no hdcp*<CR><LF>
Update Info			
1	Get selected target firmware version	**Command:** GET VER<CR><LF> **Return:** VER *prm*<CR><LF> **Parameter:** *prm* = {…}// according to actual firmware version **Description:** Get selected target firmware version.	**Command:** GET VER<CR><LF> **Return:** VER ARM VER V1.0.3 MCU VER V1.0.0 **Description:** Get all module firmware version.
2	Get selected target hardware version	**Command:** GET HW_VER<CR><LF> **Return:** HWVER *prm*<CR><LF> **Parameter:** *prm* = {…}// according to actual firmware version **Description:** Get selected target firmware version.	**Command:** GET HW_VER<CR><LF> **Return:** HW_VER V0.1 **Description:** Get all module firmware version.

No.	Description	Command	Example
		Preset Info	
1	Save Preset Scene	**Command:** SAVE PRESET *prm*<CR><LF> **Return:** PRESET *prm*<CR><LF> **Parameter:** *prm* = {1,2,3}// **Description:** Save Preset Scene.	**Command:** SAVE PRESET *1*<CR><LF> **Return:** PRESET *1* <CR><LF> **Description:** Save preset scene.
2	Restore Preset Scene	**Command:** RESTORE PRESET *prm*<CR><LF> **Return:** PRESET *prm*<CR><LF> **Parameter:** *prm* = {1,2,3}// **Description:** Restore Preset Scene.	**Command:** RESTORE PRESET *1*<CR><LF> **Return:** PRESET *1*<CR><LF> **Description:** Restore preset scene.

No.	Description	Command	Example
		Video Output	
1	Set the scaling mode of the video output port	**Command:** SET VIDOUT_SCALE *out prm*<CR><LF> **Return:** VIDOUT_SCALE *out prm*<CR><LF> **Description:** *out = {out1, out2…out4, all};// bypass only for out1 out2* *prm = {auto, manual, bypass}* *§ • auto -- which matches TV EDID automatically* *• manual --Manual change scalar output resolution* *• bypass -- It will bypass all HDMI source signal to sink display*	**Command:** SET VIDOUT_SCALE *out1 auto*<CR><LF> **Return:** VIDOUT_SCALE *out1 auto*<CR><LF> **Description:** Set Scale mode to auto.
2	Get the scaling mode of the video output port	**Command:** GET VIDOUT_SCALE *out*<CR><LF> **Return:** VIDOUT_SCALE *out prm*<CR><LF> **Description:** *out = {out1, out2…out4, all};// bypass only for out1 out2* *prm = {auto, manual, bypass}*	**Command:** GET VIDOUT_SCALE *out1*<CR><LF> **Return:** VIDOUT_SCALE *out1 auto*<CR><LF> **Description:** out1 Scale mode is auto.

No.	Description	Command	Example
3	Set output resolution	**Command:** SET VIDOUT_RES *out prm*<CR><LF> **Return:** VIDOUT_RES *out prm*<CR><LF> **Description:** This function can only be set in SCALE manual mode. *out = {out1,out2…out4,all};* *prm =* { 4096x2160@60, 4096x2160@30, 4096x2160@25, 4096x2160@24, 3840x2160@60, 3840x2160@50, 3840x2160@30, 3840x2160@25, 3840x2160@24, 1920x1200@60, 1920x1080@60, 1920x1080@50, 1280x720@60, 1280x720@50, 1680x1050@60, 1600x1200@60, 1600x900@60, 1440x900@60, 1366x768@60, 1360x768@60, 1280x1024@60, 1280x960@60, 1280x800@60, 1280x768@60, 1024x768@60, 800x600@60 , }	**Command:** SET VIDOUT_RES *out1 3840x2160@60*<CR><LF> **Return:** VIDOUT_RES *out1 3840x2160@60*<CR><LF> **Description:** Set out1 resolution to 3840x2160@60.

No.	Description	Command	Example
4	Get output resolution	**Command:** GET VIDOUT_RES *out*<CR><LF> **Return:** VIDOUT_RES *out prm*<CR><LF> **Description:** *out = {out1, out2…out4, all};* *prm=* { 4096x2160@60, 4096x2160@30, 4096x2160@25, ... 1280x800@60, 1280x768@60, 1024x768@60, 800x600@60 }	**Command:** GET VIDOUT_RES *out1*<CR><LF> **Return:** VIDOUT_RES *out1 4096x2160@60*<CR><LF> **Description:** HDMI out1's resolution is 4096x2160@60.
5	Set Output forced SDR	**Command:** SET FORCED_SDR out prm<CR><LF> **Return:** FORCED_SDR out prm<CR><LF> **Description:** out = {all}; prm = {on, off}	**Command:** SET FORCED_SDR all on<CR><LF> **Return:** FORCED_SDR out1 on<CR><LF> FORCED_SDR out2 on<CR><LF> FORCED_SDR out3 on<CR><LF> FORCED_SDR out4 on<CR><LF>
6	Get Output forced SDR	**Command:** GET FORCED_SDR out<CR><LF> **Return:** FORCED_SDR out prm<CR><LF> **Description:** out = {all};	**Command:** GET FORCED_SDR all<CR><LF> **Return:** FORCED_SDR out1 on<CR><LF> FORCED_SDR out2 on<CR><LF> FORCED_SDR out3

No.	Description	Command	Example
		prm = {on, off}	on<CR><LF> FORCED_SDR out4 on<CR><LF>