2x4 Video Wall Processor

4KVW24

API Command Set

AV Access

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*

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.

No.	Description	Command	Example
		Video Wall	
1	Set video wall mode	**Command:** SET VIDWALL prm <CR><LF> **Return:** VIDWALL prm<CR><LF> **Parameter:** prm = {quick, standard}	**Command:** SET VIDWALL quick<CR><LF> **Return:** VIDWALL quick<CR><LF>
2	Get video wall mode	**Command:** GET VIDWALL <CR><LF> **Return:** VIDWALL prm<CR><LF> **Parameter:** prm = {quick, standard}	**Command:** GET VIDWALL<CR><LF> **Return:** VIDWALL quick<CR><LF>
3	Set quick video wall mode	**Command:** SET VW_QUICK prm <CR><LF> **Return:** VW_QUICK prm<CR><LF> **Parameter:** prm = {1x2, 1x2R, 1x3, 1x4, 1x4R, 2x1, 2x2, 2x2R, SM, 3x1, 4x1}	**Command:** SET VW_QUICK 2x2<CR><LF> **Return:** VW_QUICK 2x2<CR><LF>
4	Get quick video wall mode	**Command:** GET VW_QUICK <CR><LF> **Return:** VW_QUICK prm<CR><LF> **Parameter:** prm = {1x2, 1x2R, 1x3, 1x4, 1x4R, 2x1, 2x2, 2x2R, SM, 3x1, 4x1}	**Command:** SET VW_QUICK<CR><LF> **Return:** VW_QUICK 2x2<CR><LF>

No.	Description	Command	Example
5	Set video wall input source	**Command:** SET VW_SRC prm <CR><LF> **Return:** VW_SRC prm<CR><LF> **Parameter:** prm = {in1, in2}	**Command:** SET VW_SRC in1<CR><LF> **Return:** VW_SRC in1<CR><LF>
6	Get video wall input source	**Command:** GET VW_SRC <CR><LF> **Return:** VW_SRC prm<CR><LF> **Parameter:** prm = {in1, in2}	**Command:** GET VW_SRC<CR><LF> **Return:** VW_SRC in1<CR><LF>
Video			
1	Set auto switch function to on/off	**Command:** SET AUTOSW *prm*<CR><LF> **Return:** AUTOSW *prm*<CR><LF> **Parameter:** *prm* = {on, off}; **Description:** Set auto switch function to on/off.	**Command:** SET AUTOSW *on*<CR><LF> **Return:** AUTOSW *on*<CR><LF> **Description:** Set auto switch function to on.

No.	Description	Command	Example
2	Get auto switch function enable status	**Command:** GET AUTOSW<CR><LF> **Return:** AUTOSW *prm*<CR><LF> **Parameter:** *prm* = {on, off}; **Description:** Get auto switch function enable status.	**Command:** GET AUTOSW<CR><LF> **Return:** AUTOSW *on*<CR><LF> **Description:** Get auto switch function enable status, the result is on.
		Audio	
1	Set HDMI audio to mute/unmute	**Command:** SET AVMUTE *out prm*<CR><LF> **Return:** AVMUTE *out prm*<CR><LF> **Parameter:** *out* = {out1, out2, out3, out4, all}; *prm* = {on, off}; **Description:** Set HDMI audio to mute/unmute.	**Command:** SET AVMUTE *out1 on*<CR><LF> **Return:** AVMUTE *out1 on*<CR><LF> **Description:** Set HDMI audio to mute.
2	Get HDMI audio mute/unmute status	**Command:** GET AVMUTE *out* <CR><LF> **Return:** AVMUTE *out prm*<CR><LF> **Parameter:** *out* = {out1, out2, out3, out4,all}; *prm* = {on, off}; **Description:** Get HDMI audio mute/unmute status.	**Command:** GET AVMUTE *out1*<CR><LF> **Return:** AVMUTE *out1 on*<CR><LF> **Description:** Get HDMI audio mute/unmute status, the result is mute.

No.	Description	Command	Example
3	Set Audio output to mute/unmute	**Command:** SET AUDIO_MUTE *prm*<CR><LF> **Return:** AUDIO_MUTE *prm*<CR><LF> **Parameter:** *prm* = {on, off}; **Description:** Set Audio output to mute/unmute.	**Command:** SET AUDIO_MUTE *on*<CR><LF> **Return:** AUDIO_MUTE *on*<CR><LF> **Description:** Set audio output to mute.
4	Get Audio Mute Status of Audio Output	**Command:** GET AUDIO_MUTE <CR><LF> **Return:** AUDIO_MUTE prm <CR><LF> **Parameter:** prm = {on, off}; **Description:** Get Audio Mute Status of Audio Output.	**Command:** GET AUDIO_MUTE <CR><LF> **Return:** AUDIO_MUTE on<CR><LF> **Description:** Get Audio Mute Status of Audio Output, the result is mute.
CEC			

No.	Description	Command	Example
1	Set display to POWER ON/OFF via CEC	**Command:** SET CEC_PWR *out prm*<CR><LF> **Return:** CEC_PWR *out prm*<CR><LF> **Parameter:** *prm* = {on, off} *out* = {out1, out2, out3, out4 ,all}; **Description:** Set display to POWER ON/OFF via CEC.	**Command:** SET CEC_PWR *out1 on*<CR><LF> **Return:** CEC_PWR *out1 on*<CR><LF> **Description:** Set display1 to POWER ON via CEC.
2	Set CEC AUTO POWER ON/OFF function to on/off	**Command:** SET AUTOCEC_FN *out prm*<CR><LF> **Return:** AUTOCEC_FN *out prm*<CR><LF> **Parameter:** *prm* = {on, off} *out* = {out1, out2, out3, out4, all}; **Description:** Set CEC AUTO POWER ON/OFF function to on/off.	**Command:** SET AUTOCEC_FN *out1 on*<CR><LF> **Return:** AUTOCEC_FN *out1 on*<CR><LF> **Description:** Set CEC AUTO POWER ON/OFF function of output1 to on.

No.	Description	Command	Example
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, out3, out4 , out5, out6, all}; **Description:** Get Sink auto power Function ON or OFF Status.	**Command:** GET AUTOCEC_FN *out1*<CR><LF> **Return:** AUTOCEC_FN *out1 on* **Description:** Get Sink auto power status, and the status is ON.
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, out3, 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 display will auto power off.

No.	Description	Command	Example
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, out3, 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.
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, all}* **Description:** HDCP_S will control source hdcp support 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.

No.	Description	Command	Example
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, 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.
EDID			
1	Set Input EDID	**Command:** SET EDID prm1 prm2<CR><LF> **Return:** EDID prm1 prm2<CR><LF> **Parameter:** prm1 = {all} prm2 = {1 ~21} 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 5400x1920p30 2.0CH PCM Audio with HDR; 16 : Fixed 5400x1920p30 2.0CH PCM Audio with SDR;	**Command:** SET EDID all *12*<CR><LF> **Return:** EDID *in1 12*<CR><LF> EDID in2 12<CR><LF> **Description:** Set in1 EDID to Fixed 4K60 2.0CH PCM Audio with SDR.

No.	Description	Command	Example
		17 : Fixed 3840x600p60 2.0CH PCM Audio with HDR; 18 : Fixed 3840x600p60 2.0CH PCM Audio with SDR; 19 : Fixed 1080p@60Hz 2.0CH PCM Audio with HDR; 20 : Fixed 1080p@60Hz 2.0CH PCM Audio with SDR; 21 : Fixed 1080p@50Hz 2.0CH PCM Audio with SDR; 99: custom edid **Description:** Set Input EDID.	
2	Get Input EDID status	**Command:** GET EDID *prm1*<CR><LF> **Return:** EDID *prm1 prm2*<CR> EDID prm1 *prm2*<CR><LF> ..... **Parameter:** *prm1 = {all}*; *prm2 = {1 ~21}* 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 5400x1920p30 2.0CH PCM Audio with HDR; 16: Fixed 5400x1920p30 2.0CH PCM Audio with SDR; 17: Fixed 3840x600p60 2.0CH PCM Audio with HDR;	**Command:** GET EDID *all*<CR><LF> **Return:** EDID in1 *01*<CR> EDID in2 *02*<CR> **Description:** Get all input EDID Status

No.	Description	Command	Example
		18: Fixed 3840x600p60 2.0CH PCM Audio with SDR; 19: Fixed 1080p@60Hz 2.0CH PCM Audio with HDR; 20: Fixed 1080p@60Hz 2.0CH PCM Audio with SDR; 21: Fixed 1080p@50Hz 2.0CH PCM Audio with SDR; **Description:** Get all input EDID Status	
		OSD	
1	Set OSD Display Status	**Command:** SET OSD prm1 <CR><LF> **Return:** OSD prm1<CR><LF> **Parameter:** prm1 = {on, off}; **Description:** on: OSD switch enable. off: OSD switch disable.	**Command:** SET OSD on <CR><LF> **Return:** OSD on <CR><LF> **Description:** Turn on OSD.
		System Information	
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> **Description:** System reboot.	**Command:** REBOOT<CR><LF> **Return:** REBOOT<CR><LF> **Description:** System reboot.

No.	Description	Command	Example
3	Get API list	**Command:** help<CR><LF> **Return:** xxxx **Description:** Get API list.	**Command:** help<CR><LF> **Return:** xxxx **Description:** Get the API list.
4	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>
5	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>

No.	Description	Command	Example
6	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>
7	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>
8	Set Standby	**Command:** STANDBY<CR><LF> **Return:** STANDBY!<CR><LF>	**Command:** STANDBY<CR><LF> **Return:** STANDBY!<CR><LF>
9	Set Wake	**Command:** WAKE<CR><LF> **Return:** WAKE!<CR><LF>	**Command:** WAKE<CR><LF> **Return:** WAKE!<CR><LF>

No.	Description	Command	Example
10	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>
11	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};* *prm = {Disconnected, Connected}* **Description:** Get the status of the video input.	**Command:** GET VIDIN_CONNECT in1<CR><LF> **Return:** VIDIN_CONNECT in1 Disconnected<CR><LF>
12	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}; *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
13	Get input video format information	**Command:** GET VIDIN_FORMAT *in*<CR><LF> **Return:** VIDIN_FORMAT i*n* *prm*<CR><LF> in = {in1,in2}; 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
14	Get hdcp version of input video	**Command:** GET VIDIN_HDCP *in*<CR><LF> **Return:** VIDIN_HDCP *in prm*<CR><LF> *in = {in1, in2};* *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>
15	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 status of the video output.	**Command:** GET VIDOUT_CONNECT out1<CR><LF> **Return:** VIDOUT_CONNECT out1 Disconnected<CR><LF>
16	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
17	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
18	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>
Firmware			
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
		Output	
1	Set the scaling mode for the video output port	**Command:** SET VIDOUT_SCALE *out prm*<CR><LF> **Return:** VIDOUT_SCALE *out prm*<CR><LF> **Description:** *out = {all};* *prm = {auto, manual}* *• auto -- which matches TV EDID automatically* *• manual --Manual change scalar output resolution*	**Command:** SET VIDOUT_SCALE *all auto*<CR><LF> **Return:** VIDOUT_SCALE *all auto*<CR><LF> **Description:** Set Scale mode to auto.
2	Get the scaling mode for the video output port	**Command:** GET VIDOUT_SCALE *out*<CR><LF> **Return:** VIDOUT_SCALE *out prm*<CR><LF> **Description:** *out = {all};* *prm = {auto, manual}*	**Command:** GET VIDOUT_SCALE *all*<CR><LF> **Return:** VIDOUT_SCALE *all auto*<CR><LF> **Description:** out1 Scale mode is auto.

No.	Description	Command	Example

3	To 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 = {all};
		prm =
		{
		4096x2160@60,
		4096x2160@30,
		3840x2160@60,
		3840x2160@30,
		1920x1080@60,
		1920x1080@50,
		1280x720@60,
		1280x720@50,
		}
			Command:
			SET VIDOUT_RES *all 3840x2160@60*<CR><LF>
			
			Return:
			VIDOUT_RES *all 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 = {all};
		prm=
		{
		4096x2160@60,
		4096x2160@30,
		3840x2160@60,
		3840x2160@30,
		1920x1080@60,
		1920x1080@50,
		1280x720@60,
		1280x720@50,
		}
			Command:
			GET VIDOUT_RES all<CR><LF>
			
			Return:
			VIDOUT_RES all *4096x2160@60*<CR><LF>
			
			Description:
			HDMI out resolution is 4096x2160@60.

2x4 4K60 视频墙处理器

4KVW24

API 命令手册

视连捷
AV Access

关于 Telnet 连接

在发送 telnet 命令前，应该将 telnet 连接到对应的设备。
telnet 命令格式如下：

telnet ip (port)
ip：设备的 IP 地址。
Port：设备的端口号，在一些 telnet 控制工具或者平台上，这个参数可能不需要。如果需要，端口号默认是 23。
例如：如果设备的 IP 地址是 192.168.11.143，
telnet 命令是：*telnet 192.168.11.143*

命令

以 SET SW in out<CR><LF>为例：

4. [SET SW]表示命令关键字，不区分大小写。
5. [in out]表示参数，不区分大小写，不正确的参数将不会被识别。
6. <CR><LF>表示回车或者换行；所有的命令必须以回车或换行结束。

序号	描述	命令	示例
		视频墙	
1	设置视频墙模式	**命令：** SET VIDWALL prm <CR><LF> **返回：** VIDWALL prm<CR><LF> **参数：** prm = {quick, standard}	**命令：** SET VIDWALL quick<CR><LF> **返回：** VIDWALL quick<CR><LF>
2	获取视频墙模式	**命令：** GET VIDWALL <CR><LF> **返回：** VIDWALL prm<CR><LF> **参数：** prm = {quick, standard}	**命令：** GET VIDWALL<CR><LF> **返回：** VIDWALL quick<CR><LF>
3	设置快速视频墙模式	**命令：** SET VW_QUICK prm <CR><LF> **返回：** VW_QUICK prm<CR><LF> **参数：** prm = {1x2, 1x2R, 1x3, 1x4, 1x4R, 2x1, 2x2, 2x2R, SM, 3x1, 4x1}	**命令：** SET VW_QUICK 2x2<CR><LF> **返回：** VW_QUICK 2x2<CR><LF>
4	获取快速视频墙模式	**命令：** GET VW_QUICK <CR><LF> **返回：** VW_QUICK prm<CR><LF> **参数：** prm = {1x2, 1x2R, 1x3, 1x4, 1x4R, 2x1, 2x2, 2x2R, SM, 3x1, 4x1}	**命令：** SET VW_QUICK<CR><LF> **返回：** VW_QUICK 2x2<CR><LF>

序号	描述	命令	示例
5	设置视频墙输入源	**命令：** SET VW_SRC prm <CR><LF> **返回：** VW_SRC prm<CR><LF> **参数：** prm = {in1, in2}	**命令：** SET VW_SRC in1<CR><LF> **返回：** VW_SRC in1<CR><LF>
6	获取视频墙输入源	**命令：** GET VW_SRC <CR><LF> **返回：** VW_SRC prm<CR><LF> **参数：** prm = {in1, in2}	**命令：** GET VW_SRC<CR><LF> **返回：** VW_SRC in1<CR><LF>
视频			
1	设置自动切换功能开启/关闭	**命令** SET AUTOSW *prm*<CR><LF> **返回** AUTOSW *prm*<CR><LF> **参数：** *prm* = {on, off}; **描述：** 设置自动切换功能开启/关闭。	**命令** SET AUTOSW *on*<CR><LF> **返回** AUTOSW *on*<CR><LF> **描述：** 设置开启自动切换功能。

序号	描述	命令	示例
2	获取自动切换功能开启/关闭状态	**命令** GET AUTOSW<CR><LF> **返回** AUTOSW *prm*<CR><LF> **参数：** *prm* = {on, off}; **描述：** 获取自动切换功能开启/关闭状态	**命令** GET AUTOSW<CR><LF> **返回** AUTOSW *on*<CR><LF> **描述：** 获取自动切换功能开启/关闭状态，结果是开启。
音频			
1	设置 HDMI 音频静音/取消静音	**命令** SET AVMUTE *out prm*<CR><LF> **返回** AVMUTE *out prm*<CR><LF> **参数：** *out* = {out1, out2, out3, out4, all}; *prm* = {on, off}; **描述：** 设置 HDMI 音频静音/取消静音。	**命令** SET AVMUTE *out1 on*<CR><LF> **返回** AVMUTE *out1 on*<CR><LF> **描述：** 设置 HDMI 音频静音/取消静音。
2	获取 HDMI 音频是否静音	**命令** GET AVMUTE *out* <CR><LF> **返回** AVMUTE *out prm*<CR><LF> **参数：** *out* = {out1, out2, out3, out4,all}; *prm* = {on, off}; **描述：** 获取 HDMI 音频是否静音。	**命令** GET AVMUTE *out1*<CR><LF> **返回** AVMUTE *out1 on*<CR><LF> **描述：** HDMI 音频静音。

序号	描述	命令	示例
3	设置 Audio Out 静音/不静音	**命令** SET AUDIO_MUTE *prm*<CR><LF> **返回** AUDIO_MUTE *prm*<CR><LF> **参数：** *prm* = {on, off}; **描述：** 设置 Audio Out 静音/不静音。	**命令** SET AUDIO_MUTE *on*<CR><LF> **返回** AUDIO_MUTE *on*<CR><LF> **描述：** 设置 Audio Out 静音。
4	获取 Audio Out 静音状态	**命令** GET AUDIO_MUTE <CR><LF> **返回** AUDIO_MUTE prm <CR><LF> **参数：** prm = {on, off}; **描述：** 获取 Audio Out 静音状态。	**命令** GET AUDIO_MUTE <CR><LF> **返回** AUDIO_MUTE on<CR><LF> **描述：** 获取 Audio Out 静音状态，结果是静音。

序号	描述	命令	示例
		CEC	
1	设置显示器通过 CEC 开机/关机	**命令** SET CEC_PWR *out prm*<CR><LF> **返回** CEC_PWR *out prm*<CR><LF> **参数：** *prm* = {on, off} *out* = {out1, out2, out3, out4 ,all}; **描述：** 设置显示器通过 CEC 开机/关机。	**命令** SET CEC_PWR *out1 on*<CR><LF> **返回** CEC_PWR *out1 on*<CR><LF> **描述：** 设置显示器 1 通过 CEC 开机。
2	设置 CEC 自动开关机功能开启/关闭	**命令** SET AUTOCEC_FN *out prm*<CR><LF> **返回** AUTOCEC_FN *out prm*<CR><LF> **参数：** *prm* = {on, off} *out* = {out1, out2, out3, out4, all}; **描述：** 设置 CEC 自动开关机功能开启/关闭。	**命令** SET AUTOCEC_FN *out1 on*<CR><LF> **返回** AUTOCEC_FN *out1 on*<CR><LF> **描述：** 设置 CEC 自动开关机功能开启。

序号	描述	命令	示例
3	获取 CEC 自动开关机功能开启/关闭状态	**命令** GET AUTOCEC_FN *out*<CR><LF> **返回** AUTOCEC_FN *out prm*<CR><LF> **参数：** *prm* = {on,off} *out* = {out1, out2, out3, out4 , out5, out6, all}; **描述：** 获取 CEC 自动开关机功能开启/关闭状态。	**命令** GET AUTOCEC_FN *out1*<CR><LF> **返回** AUTOCEC_FN *out1 on* **描述：** 获取 CEC 自动开关机功能开启/关闭状态，结果是开启。
4	设置 CEC 自动关机延迟时间	**命令** SET AUTOCEC_D *out prm*<CR><LF> **返回** AUTOCEC_D *out prm*<CR><LF> **参数：** *out* = {out1, out2, out3, out4, all}; *prm* = {1~30}// 1 的意思是 1 分钟，2 的意思是 2 分钟，默认是 2 分钟，最大延迟时间为 30 分钟 **描述：** AUTOCEC_D 是 CEC auto Power Delay Timing 的简写。	**命令** SET AUTOCEC_D *out1 2*<CR><LF> **返回** AUTOCEC_D *out1 2*<CR><LF> **描述：** 当 HDMI out 1 没有活跃信号时，2 分钟后，连接诶的显示器将会自动关闭。

序号	描述	命令	示例
5	获取 CEC 自动关机的延迟时间	**命令** GET AUTOCEC_D *out*<CR><LF> **返回** AUTOCEC_D *out prm*<CR><LF> **参数：** *out* = {out1, out2, out3, out4, all}; *prm* = {1~30}// 1 的意思是 1 分钟，2 的意思是 2 分钟，默认是 2 分钟，最大延迟时间为 30 分钟 **描述：** AUTOCEC_D 是 CEC auto Power Delay Timing 的简写。	**命令** GET AUTOCEC_D *out1*<CR><LF> **返回** AUTOCEC_D *out1 2*<CR><LF> **描述：** HDMI out1 的 CEC 关机自动延迟时间是 2 分钟。
HDCP			
1	设置输入启用/关闭 HDCP	**命令** SET HDCP_S *in prm*<CR><LF> **返回** HDCP_S *in prm*<CR><LF> **参数：** *prm* = {on, off} *in = {in1, in2, all}* **描述：** 设置输入启用/关闭 HDCP。	**命令** SET HDCP_S *in1 on*<CR><LF> **返回** HDCP_S *in1 on*<CR><LF> **描述：** 输入 1 的 HDCP 设置为开启。

序号	描述	命令	示例
2	获取输入 HDCP 的状态	**命令** GET HDCP_S *in* <CR><LF> **返回** HDCP_S *in prm*<CR><LF> **参数：** *prm* = {on, off} *in = {in1, in2, all}* **描述：** HDCP_S 是 HDCP support 的简写。	**命令** GET HDCP_S in1<CR><LF> **返回** HDCP_S *in1 on*<CR><LF> **描述：** 输入 1 的 HDCP 设置为开启。
EDID			
1	设置输入 EDID	**命令** SET EDID prm1 prm2<CR><LF> **返回** EDID prm1 prm2<CR><LF> **参数：** prm1 = {all} prm2 = {1 ~21} 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 5400x1920p30 2.0CH PCM Audio with HDR; 16： Fixed 5400x1920p30 2.0CH PCM Audio with SDR;	**命令** SET EDID all *12*<CR><LF> **返回** EDID *in1 12*<CR><LF> EDID in2 12<CR><LF> **描述：** 设置输入 1 的 EDID 为 Fixed 4K60 2.0CH PCM Audio with SDR。

序号	描述	命令	示例
		17 : Fixed 3840x600p60 2.0CH PCM Audio with HDR; 18 : Fixed 3840x600p60 2.0CH PCM Audio with SDR; 19 : Fixed 1080p@60Hz 2.0CH PCM Audio with HDR; 20 : Fixed 1080p@60Hz 2.0CH PCM Audio with SDR; 21 : Fixed 1080p@50Hz 2.0CH PCM Audio with SDR; 99: custom edid **描述：** 设置输入 EDID。	
2	获取输入 EDID	**命令** GET EDID *prm1*<CR><LF> **返回** EDID *prm1 prm2*<CR> EDID prm1 *prm2*<CR><LF> ..... **参数：** *prm1 = {all};* *prm2 = {1 ~21}* 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 5400x1920p30 2.0CH PCM Audio with HDR; 16: Fixed 5400x1920p30 2.0CH PCM Audio with SDR; 17: Fixed 3840x600p60 2.0CH PCM Audio with HDR;	**命令** GET EDID *all*<CR><LF> **返回** EDID in1 *01*<CR> EDID in2 *02*<CR> **描述：** 获取所有输入的 EDID。

序号	描述	命令	示例
		18: Fixed 3840x600p60 2.0CH PCM Audio with SDR; 19: Fixed 1080p@60Hz 2.0CH PCM Audio with HDR; 20: Fixed 1080p@60Hz 2.0CH PCM Audio with SDR; 21: Fixed 1080p@50Hz 2.0CH PCM Audio with SDR; **描述：** 获取输入 EDID。	
		OSD	
1	设置 OSD 显示状态	**命令** SET OSD prm1 <CR><LF> **返回** OSD prm1<CR><LF> **参数：** prm1 = {on, off}; **描述：** on: 启用 OSD。 off: 关闭 OSD。	**命令** SET OSD on <CR><LF> **返回** OSD on <CR><LF> **描述：** 开启 OSD。
		系统信息	
1	恢复出厂设置	**命令** RESET<CR><LF> **返回** RESET<CR><LF> **描述：** 恢复出厂设置。	**命令** RESET<CR><LF> **返回** RESET<CR><LF> **描述：** 恢复出厂设置。
2	重启设备。	**命令** REBOOT<CR><LF> **返回** REBOOT<CR><LF> **描述：** 重启设备。	**命令** REBOOT<CR><LF> **返回** REBOOT<CR><LF> **描述：** 重启设备。

序号	描述	命令	示例
3	获取 API 列表	**命令** help<CR><LF> **返回** xxxx **描述：** 获取 API 列表。	**命令** help<CR><LF> **返回** xxxx **描述：** 获取 API 列表。
4	获取 IP 地址	**命令** GET IPADDR<CR><LF> **返回** IPADDR xxx.xxx.xxx.xxx MASK xxx.xxx.xxx.xxx GATEWAY xxx.xxx.xxx.xxx<CR><LF> **描述：** 获取 IP 地址。	**命令** GET IPADDR<CR><LF> **返回** IPADDR 192.168.1.2 MASK 255.255.255.0 GATEWAY 192.168.2.1<CR><LF>
5	设置 IP 地址	**命令** SET IPADDR xxx.xxx.xxx.xxx MASK xxx.xxx.xxx.xxx GATEWAY xxx.xxx.xxx.xxx<CR><LF> **返回** IPADDR xxx.xxx.xxx.xxx MASK xxx.xxx.xxx.xxx GATEWAY xxx.xxx.xxx.xxx<CR><LF> **描述：** 设置 IP 地址。	**命令** SET IPADDR 192.168.1.2 MASK 255.255.255.0 GATEWAY 192.168.2.1<CR><LF> **返回** IPADDR 192.168.1.2 MASK 255.255.255.0 GATEWAY 192.168.2.1<CR><LF>

序号	描述	命令	示例
6	获取 IP 模式	**命令** GET NETCFG MODE<CR><LF> **返回** NETCFG MODE prm<CR><LF> **参数：** prm = {DHCP, STATIC}; **描述：** 获取 IP 模式。	**命令** GET NETCFG MODE<CR><LF> **返回** NETCFG MODE DHCP<CR><LF>
7	设置 IP 模式	**命令** SET NETCFG MODE prm<CR><LF> **返回** NETCFG MODE prm<CR><LF> **参数：** prm = {DHCP, STATIC}; **描述：** 设置 IP 模式。	**命令** SET NETCFG MODE DHCP<CR><LF> **返回** NETCFG MODE DHCP<CR><LF>
8	设置待机	**命令** STANDBY<CR><LF> **返回** STANDBY!<CR><LF>	**命令** STANDBY<CR><LF> **返回** STANDBY!<CR><LF>
9	设置唤醒	**命令** WAKE<CR><LF> **返回** WAKE!<CR><LF>	**命令** WAKE<CR><LF> **返回** WAKE!<CR><LF>

序号	描述	命令	示例
10	获取待机状态	**命令** GET STANDBY<CR><LF> **返回** STANDBY!<CR><LF> or WAKE!<CR><LF>	**命令** GET STANDBY<CR><LF> **返回** STANDBY!<CR><LF> or WAKE!<CR><LF>
11	获取视频输入的连接状态	**命令** GET VIDIN_CONNECT *in*<CR><LF> **返回** VIDIN_CONNECT *in prm*<CR><LF> *in = {in1, in2};* *prm = {Disconnected, Connected}* **描述：** 获取视频输入的连接状态。	**命令** GET VIDIN_CONNECT in1<CR><LF> **返回** VIDIN_CONNECT in1 Disconnected<CR><LF>
12	获取视频输入的信号状态	**命令** GET VIDIN_SIG *in*<CR><LF> **返回** VIDIN_SIG *in prm*<CR><LF> *in* = {in1,in2}; *prm* = {no, valid} **描述：** 获取视频输入的信号状态。	**命令** GET VIDIN_SIG in1<CR><LF> **返回** VIDIN_SIG in1 valid<CR><LF>

序号	描述	命令	示例
13	获取输入视频格式	**命令** GET VIDIN_FORMAT *in*<CR><LF> **返回** VIDIN_FORMAT i*n* *prm*<CR><LF> in = {in1,in2}; prm = { } **描述：** 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	**命令** GET VIDIN_FORMAT *in1*<CR><LF> **返回** VIDIN_FORMAT *in1* *3840*x2160,60;none hdr;rgb;8bit<CR><LF>

序号	描述	命令	示例
14	获取输入视频的 HDCP 版本	**命令** GET VIDIN_HDCP *in*<CR><LF> **返回** VIDIN_HDCP *in prm*<CR><LF> *in = {in1, in2};* *prm = {no hdcp, hdcp1.4, hdcp2.2}* **描述：** 获取输入视频的 HDCP 版本。	**命令** GET VIDIN_HDCP *in1*<CR><LF> **返回** VIDIN_HDCP *in1 no hdcp*<CR><LF>
15	获取视频输出的连接	**命令** GET VIDOUT_CONNECT *out*<CR><LF> **返回** VIDOUT_CONNECT *in prm*<CR><LF> *out = {out1, out2...out4}*; *prm = {Disconnected, Connected}* **描述：** Get the status of the video output.	**命令** GET VIDOUT_CONNECT out1<CR><LF> **返回** VIDOUT_CONNECT out1 Disconnected<CR><LF>
16	获取视频输出的信号状态	**命令** GET VIDOUT_SIG *out*<CR><LF> **返回** VIDOUT_SIG *out prm*<CR><LF> *out* = {out1, out2…out4}; *prm* = {no, valid} **描述：** 获取视频输出的信号状态。	**命令** GET VIDOUT_SIG out1<CR><LF> **返回** VIDOUT_SIG *out1 valid*<CR><LF>

序号	描述	命令	示例
17	获取输出的视频格式	**命令** GET VIDOUT_FORMAT *out*<CR><LF> **返回** VIDOUT_FORMAT out *prm*<CR><LF> *out = {out1, out2…out4};* *prm = { }* **描述：** 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	**命令** GET VIDOUT_FORMAT *out1*<CR><LF> **返回** VIDOUT_FORMAT *out1 3840*x2160,60;none hdr;rgb;8bit<CR><LF>

序号	描述	命令	示例
18	获取输出视频的 HDCP 版本	**命令** GET VIDOUT_HDCP *out*<CR><LF> **返回** VIDOUT_HDCP *out prm*<CR><LF> *out = {out1,out2…out4};* *prm = {no hdcp, hdcp1.4, hdcp2.2}* **描述：** 获取输出视频的 HDCP 版本。	**命令** GET VIDOUT_HDCP *out1*<CR><LF> **返回** VIDOUT_HDCP *out1 no hdcp*<CR><LF>
固件			
1	获取固件版本	**命令** GET VER<CR><LF> **返回** VER *prm*<CR><LF> **参数：** *prm* = {…}// according to actual firmware version **描述：** 获取固件版本。	**命令** GET VER<CR><LF> **返回** VER ARM VER V1.0.3 MCU VER V1.0.0 **描述：** 获取固件版本。
2	设置硬件版本	**命令** GET HW_VER<CR><LF> **返回** HWVER *prm*<CR><LF> **参数：** *prm* = {…}// according to actual firmware version **描述：** 设置硬件版本。	**命令** GET HW_VER<CR><LF> **返回** HW_VER V0.1 **描述：** 设置硬件版本。

序号	描述	命令	示例
		输出	
1	设置视频输出端口的倍线模式	**命令** SET VIDOUT_SCALE *out prm*<CR><LF> **返回** VIDOUT_SCALE *out prm*<CR><LF> **描述：** *out = {all};* *prm = {auto, manual}* *• auto – 自动匹配电视的 EDID* *• manual – 手动改变 scaler 的分辨率*	**命令** SET VIDOUT_SCALE *all auto*<CR><LF> **返回** VIDOUT_SCALE *all auto*<CR><LF> **描述：** 设置所有输出的倍线模式为自动。
2	获取视频输出端口的倍线模式	**命令** GET VIDOUT_SCALE *out*<CR><LF> **返回** VIDOUT_SCALE *out prm*<CR><LF> **描述：** *out = {all};* *prm = {auto, manual}*	**命令** GET VIDOUT_SCALE *all*<CR><LF> **返回** VIDOUT_SCALE *all auto*<CR><LF> **描述：** 输出 1 的 Scale 模式设置为自动。

序号	描述	命令	示例

3	设置输出分辨率	**命令**
		SET VIDOUT_RES *out prm*<CR><LF>
		
		返回
		VIDOUT_RES *out prm*<CR><LF>
		
		描述：
		此功能只能在手动倍线模式下设置
		out = {all};
		prm =
		{
		4096x2160@60,
		4096x2160@30,
		3840x2160@60,
		3840x2160@30,
		1920x1080@60,
		1920x1080@50,
		1280x720@60,
		1280x720@50,
		}
			命令
			SET VIDOUT_RES *all 3840x2160@60*<CR><LF>
			
			返回
			VIDOUT_RES *all 3840x2160@60*<CR><LF>
			
			描述：
			设置输出 1 的分辨率为 3840x2160@60。

序号	描述	命令	示例

4	获取输出分辨率	**命令**
		GET VIDOUT_RES *out*<CR><LF>
		
		返回
		VIDOUT_RES *out prm*<CR><LF>
		
		描述：
		out = {all};
		prm=
		{
		4096x2160@60,
		4096x2160@30,
		3840x2160@60,
		3840x2160@30,
		1920x1080@60,
		1920x1080@50,
		1280x720@60,
		1280x720@50,
		}
			命令
			GET VIDOUT_RES all<CR><LF>
			
			返回
			VIDOUT_RES all *4096x2160@60*<CR><LF>



AV Access