API Command Set_ SCT-MXKVM42-H2U3

HDMI2.0 KVM Matrix Switcher

Version: V1.0.7

RS232 Default Setting

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

About Telnet Connection

Before the process of sending the telnet command, shall make telnet connection to the corresponding device.

The form of telnet command is as follow:

telnet ip (port)

ip: The unit's IP address.

port: The unit's port number, this is non-required on some Telnet control tools or platforms. If required, port number is 23 by default.

Example: If the unit's IP address is 192.168.11.143,

The telnet command is *telnet 192.168.11.143*

Command

Take Command SET SW in out<CR><LF> as an example:

1. [SET SW] denotes command key words, case insensitive.
2. [in out] 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
Normal switch case			
1	Switch one input to one output	**Command:** SET VIDSW *out in*<CR><LF> **Return:** VIDSW out in<CR><LF> **Parameter:** *in* = {i00, i01, i02, i03, i04}; *out* = {o01, o02}; **Description:** SW is short for Switch Switch one input source to one output sink. i00 is none input.	**Command:** SET VIDSW o01 i02<CR><LF> **Return:** VIDSW o01 i02<CR><LF> **Description:** Switch input 2 to hdmi output 1.
2	Switch one input to all outputs	**Command:** SET VIDSW all in<CR><LF> **Return:** VIDSW *all in*<CR><LF> **Parameter:** *in* = {i00, i01, i02, i03, i04}; *all* = {all}; **Description:** SW is short for Switch. Switch one input source to all output sinks. i00 is none input.	**Command:** SET VIDSW all i01 <CR><LF> **Return:** VIDSW all i01<CR><LF> **Description:** Switch input1 to all output sinks.

No.	Description	Command	Example
3	Get which input is mapping to the indicate Output	**Command:** GET VIDSW *out*<CR><LF> **Return:** VIDSW *out in*<CR><LF> **Parameter:** in = {i00, i01, i02, i03, i04}; out = {o01, o02}; **Description:** Get which input is mapping to the indicate Output. i00 is none input.	**Command:** GET VIDSW *o02*<CR><LF> **Return:** VIDSW o02 i01<CR><LF> **Description:** Get which input is mapping to output 2.
4	Get the correspondence among all inputs and outputs	**Command:** GET VIDSW *all*<CR><LF> **Return:** VIDSW *out in*<CR><LF> VIDSW *out in*<CR><LF> ..... **Parameter:** in = {i00, i01, i02, i03, i04}; out = {o01, o02}; *all* = {all}; **Description:** Get the correspondence among all inputs and outputs. i00 is none input.	**Command:** GET VIDSW *all* <CR><LF> **Return:** VIDSW *o01 i01*<CR><LF> VIDSW *o02 i01*<CR><LF> ... **Description:** Get the correspondence among all inputs and outputs.

No.	Description	Command	Example
5	Switch the input source for the RX connected with the HDBT Out port between HDBT In and HDMI In	**Command:** SET VIDSWRX *out in*<CR><LF> **Return:** VIDSWRX out in<CR><LF> **Parameter:** *out* = {o01, o02, all}; in = {i01, i02}; **Description:** out= {o01=hdbtout1, o02=hdbtout2, all=both outputs} in = {i01=HDBT in, i02= HDMI in}	**Command:** SET VIDSWRX o01 i02<CR><LF> **Return:** VIDSW o01 i02<CR><LF> **Description:** Switch the input source of the RX connected with the HDBT Out 1 port to local HDMI In.
6	Get current input source (HDBT In or HDMI In) the RX connected with HDBT port selected	**Command:** GET VIDSWRX *out*<CR><LF> **Return:** VIDSWRX *out in*<CR><LF> **Parameter:** out = {o01, o02, all}; in = {i01, i02}; **Description:** out={o01=hdbtout1, o02=hdbtout2, all=both outputs} in = {i01=HDBT in, i02= HDMI in}	**Command:** GET VIDSWRX *o02*<CR><LF> **Return:** VIDSWRX o02 i02<CR><LF> **Description:** The current input source the RX connected with HDBT Out 2 port selected is local HDMI In.

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* = {o01, o02, o03, o04, all}; **Description:** Set sink power on or off. out= {o01=hdmiout1, o02=hdmiout2, o03=hdbtout1, o04=hdbtout2} all = {all};	**Command:** SET CEC_PWR o01 *on*<CR><LF> **Return:** CEC_PWR o01 *on*<CR><LF> **Description:** Set sink hdmi output 1 power on.

No.	Description	Command	Example
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* = {o01, o02, o03, o04, all}; **Description:** Set sink auto power Function ON or OFF. out= {o01=hdmiout1, o02=hdmiout2, o03=hdbtout1, o04=hdbtout2} all = {all}; **Default:** on	**Command:** SET AUTOCEC_FN *o01 on*<CR><LF> **Return:** AUTOCEC_FN o01 *on*<CR><LF> **Description:** Set sink hdmi output 1 auto power 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* = {o01, o02, o03, o04, all}; **Description:** Get Sink auto power Function ON or OFF Status. out= {o01=hdmiout1, o02=hdmiout2, o03=hdbtout1, o04=hdbtout2} all = {all}; **Default:** on	**Command:** GET AUTOCEC_FN o01<CR><LF> **Return:** AUTOCEC_FN *o01 on*<CR><LF> **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* = {o01, o02, o03, o04, all}; *prm* = {1,2,3…,}// 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. out={o01=hdmiout1, o02=hdmiout2, o03=hdbtout1, o04=hdbtout2} all={all};	**Command:** SET AUTOCEC_D *o01 2*<CR><LF> **Return:** AUTOCEC_D o01 *2*<CR><LF> **Description:** when no active signal to hdmi1, 2 minutes later, the unit will auto power off.

No.	Description	Command	Example
5	Get CEC POWER Delay Time	**Command:** GET AUTOCEC_D *out*<CR><LF> **Return:** AUTOCEC_D *out prm*<CR><LF> **Parameter:** *out* = {o01, o02, o03, o04, all}; *prm* = {1,2,3…,}// 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. out={o01=hdmiout1, o02=hdmiout2, o03=hdbtout1, o04=hdbtout2} all = {all}; **Default:** 2	**Command:** GET AUTOCEC_D o01<CR><LF> **Return:** AUTOCEC_D *o01 2*<CR><LF> **Description:** Get hdmi1 auto power delay time, the result is 2 minutes.

No.	Description	Command	Example
		HDCP	
1	Set Input HDCP to ON/OFF	**Command:** SET HDCP_S *in prm*<CR><LF> **Return:** HDCP_S *in prm*<CR><LF> **Parameter:** *prm* = {on, off} *in* = {i01, i02, i03, i04}; **Description:** HDCP_S will control source hdcp support on or off. **Default:** on	**Command:** SET HDCP_S i01 *on*<CR><LF> **Return:** HDCP_S i01 *on*<CR><LF> **Description:** Set hdmi input 1 hdcp support on.
2	Get Input HDCP ON/OFF Status	**Command:** GET HDCP_S *in*<CR><LF> **Return:** HDCP_S *in prm*<CR><LF> **Parameter:** *prm* = {on, off} *in* = {i01, i02, i03, i04}; **Description:** HDCP_S is short for HDCP support. **Default:** on	**Command:** GET HDCP_S i01<CR><LF> **Return:** HDCP_S *i01 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* = {i01,i02,i03,i04}; prm = {1 ~07} 01: Copy form HDMIoutput 1 02: Copy form HDMIoutput 2 03: Copy form HDBT output 1 04: Copy form HDBT output 2 05: 4K@60Hz 2.0ch PCM With SDR 06: 4K@30Hz 2.0ch PCM With SDR 07: 1080P@60Hz 2.0ch PCM With SDR ... **Description:** Set Input EDID. default EDID is 4K60 444 2.0ch PCM With SDR.	**Command:** SET EDID i01 *06*<CR><LF> **Return:** EDID i01 *06*<CR><LF> **Description:** Set in1 EDID Fix 1080P@60Hz 2.0ch PCM With SDR.

No.	Description	Command	Example
2	Get All Input EDID	**Command:** GET EDID *all*<CR><LF> **Return:** EDID *in prm*<CR> EDID *in prm*<CR> EDID *in prm*<CR><LF> ..... **Parameter:** *in* = {i01, i02, i03, i04}; prm = {1 ~8} 01: Copy form HDMI output 1 02: Copy form HDMI output 2 03: Copy form HDBT output 1 04: Copy form HDBT output 2 05: 4K@60Hz 2.0ch PCM With SDR 06: 4K@30Hz 2.0ch PCM With SDR 07: 1080P@60Hz 2.0ch PCM With SDR ... 08 : EDID Write **Description:** Get all input EDID. **Default:** Default EDID is 4K60 444 2.0ch PCM With SDR.	**Command:** GET EDID *all* <CR><LF> **Return:** EDID i01 *01*<CR> EDID i02 *02*<CR> EDID i03 *03*<CR><LF> …… **Description:** Get all input EDID.

No.	Description	Command	Example
3	Get one input's EDID	**Command:** GET EDID *in*<CR><LF> **Return:** EDID *in prm*<CR><LF> **Parameter:** *in* = {i01, i02, i03, i04}; prm = {1 ~8} 01: Copy form HDMI output 1 02: Copy form HDMI output 2 03: Copy form HDBT output 1 04: Copy form HDBT output 2 05: 4K@60Hz 2.0ch PCM With SDR 06: 4K@30Hz 2.0ch PCM With SDR 07: 1080P@60Hz 2.0ch PCM With SDR ... 08: EDID Write **Description:** Get one input's EDID. **Default:** defualt EDID is 4K60 444 2.0ch PCM With SDR.	**Command:** GET EDID i01<CR><LF> **Return:** EDID i01 *06*<CR><LF> **Description:** Get in1 edid status, and the status is Fix 1080P@60Hz 2.0ch PCM With SDR.

No.	Description	Command	Example
5	Set Write Input EDID.	**Command:** SET EDID_W *in prm1 prm2*<CR><LF> **Return:** EDID_W *in prm1 prm3*<CR><LF> **Parameter:** in = {i01, i02, i03, i04}; *prm1* = {block0, block1}; *prm2* = one block of 256 bytes edid ascii data with no spaces (hex data need conversion into ASCII code) *prm3* = {ok, error}; error: check sum error **Description:** Write EDID content to one input.	**Command:** SET EDID_W *i01 block0 XX…XX*<CR><LF> **Return:** EDID_W *i01 block0 ok*<CR><LF> **Description:** Write EDID content to one input.

No.	Description	Command	Example
6	Get Read Output EDID	**Command:** GET EDID_R *out*<CR><LF> **Return:** EDID_R *out prm1 prm2*<CR><LF> **Parameter:** *out* = {o01, o02, o03, o04}; *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. out={o01=hdmiout1, o02=hdmiout2, o03=hdbtout1, o04=hdbtout2}	**Command:** GET EDID_R *o01*<CR><LF> **Return:** EDID_R *o01 block0 XX…XX*<CR><LF> EDID_R *o01 block1 XX…XX*<CR><LF> **Description:** EDID_R *o01 block0 XX…XX*<CR><LF> --- Read EDID ok or EDID_R *o01 error*<CR><LF> --- Check Sum Error or EDID_R *o01 unconnect*<CR><LF> --- Sink unconnect
System Info			
1	Standby	**Command:** Standby<CR><LF> **Return:** Standby!<CR><LF> **Description:** Enter Standby mode.	**Command:** Standby<CR><LF> **Return:** Standby!<CR><LF>
2	Wake	**Command:** Wake<CR><LF> **Return:** Wake!<CR><LF>	**Command:** Wake<CR><LF> **Return:** Wake!<CR><LF>

No.	Description	Command	Example
		Description: Exit Standby mode.	
3	Get the device's standby status	**Command:** GET STANDBY<CR><LF> **Return:** STANDBY! Or WAKE!<CR><LF> Parameter: **Description:** Get the device's standby status.	**Command:** GET STANDBY<CR><LF> **Return:** WAKE!<CR><LF> **Description:** Get the device's standby status.
4	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 board.
5	System reboot	**Command:** REBOOT<CR><LF> **Return:** REBOOT<CR><LF> **Description:** System reboot.	**Command:** REBOOT<CR><LF> **Return:** REBOOT<CR><LF> **Description:** System reboot.
6	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
7	Set IP Mode	**Command:** SET IP MODE prm<CR><LF> **Return:** IP MODE prm <CR><LF> **Parameter:** pcm= {static, dhcp} **Description:** Set IP mode. **Default:** DHCP	**Command:** SET IP MODE dhcp<CR><LF> **Return:** IP MODE dhcp [<CR><LF> **Description:** Set IP mode to dhcp.
8	Get IP Mode	**Command:** GET IP MODE<CR><LF> **Return:** IP MODE prm <CR><LF> **Parameter:** pcm= {static, dhcp} **Description:** Get IP mode. **Default:** DHCP	**Command:** GET IP MODE<CR><LF> **Return:** IP MODE dhcp <CR><LF> **Description:** IP mode is dhcp.

No.	Description	Command	Example
8	SET IP address	**Command:** SET IPADDR xx.xx.xx.xx xx.xx.xx.xx xx.xx.xx.xx<CR><LF> **Return:** IPADDR IP:xx.xx.xx.xx MASK: xx.xx.xx.xx GATE: xx.xx.xx.xx<CR><LF> **Description:** SET IP address.	**Command:** SET IPADDR 192.168.1.4 255.255.255.0 192.168.1.1<CR><LF> **Return:** IPADDR IP:192.168.1.4 MASK:255.255.255.0 GATE:192.168.1.1[<CR><LF> **Description:** Set IP address is 192.168.1.4, MASK is 255.255.255.0, GATE is 192.168.1.1.
9	GET IP address	**Command:** GET IPADDR<CR><LF> **Return:** IPADDR IP:xx.xx.xx.xx MASK: xx.xx.xx.xx GATE: xx.xx.xx.xx<CR><LF> **Description:** GET IP address.	**Command:** GET IPADDR<CR><LF> **Return:** IPADDR IP:192.168.1.4 MASK:255.255.255.0 GATE:192.168.1.1<CR><LF> **Description:** Get IP address is 192.168.1.4, MASK is 255.255.255.0, GATE is 192.168.1.1.

No.	Description	Command	Example
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 1.0, ARM VER 1.0<CR><LF> **Description:** Get all module firmware version.
2	Upgrade module	**Command:** UPG [prm] <CR><LF> **Return:** UPG [prm] <CR><LF> **Parameter:** prm= {MASTER, ARM} **Description:** Upgrade module.	**Command:** UPG MASTER<CR><LF> **Return:** UPG MASTER<CR><LF> **Description:** Upgrade module.
Preset scene			
1	Save Preset Scene	**Command:** SAVE PRESET *prm*<CR><LF> **Return:** SAVE PRESET *prm*<CR><LF> **Parameter:** *prm* = {1,2,3}// **Description:** Save Preset Scene.	**Command:** SAVE PRESET *1*<CR><LF> **Return:** SAVE PRESET *1* <CR><LF> **Description:** Save preset scene.

No.	Description	Command	Example
2	Restore Preset Scene	**Command:** RESTORE PRESET *prm*<CR><LF> **Return:** RESTORE PRESET *prm*<CR><LF> **Parameter:** *prm* = {1,2,3}// **Description:** Restore Preset Scene **Default:** mp hdmi in1 out1. mp hdmi in2 out2.	**Command:** RESTORE PRESET *1*<CR><LF> **Return:** RESTORE PRESET *1*<CR><LF> **Description:** Restore preset scene.
Audio			
1	Switch one audio Input to one audio Output	**Command:** SET AUDSW aout out<CR><LF> **Return:** AUDSW aout out<CR><LF> **Parameter:** out = {o00, o01, o02}; aout = {ao01}; **Description:** Switch one input source to one output sink for audio.	**Command:** SET AUDSW ao01 o02<CR><LF> **Return:** AUDSW ao01 o02<CR><LF> **Description:** Switch audio input 2 to audio output 1.

No.	Description	Command	Example
2	Get which audio input mapping is to the indicate audio Output.	**Command:** GET AUDSW aout<CR><LF> **Return:** AUDSW aout out<CR><LF> **Parameter:** out = {o00, o01,o02}; aout = {ao01}; **Description:** Get which audio input mapping is to the indicate audio Output.	**Command:** GET AUDSW ao01<CR><LF> **Return:** AUDSW ao01 o01<CR><LF> **Description:** Get which audio input mapping is to audio Output 1.

3	Set Audio Output to mute/unmute	**Command:**
		SET MUTE a*out* *prm*<CR><LF>
		
		Return:
		MUTE a*out* *prm*<CR><LF>
		
		Parameter:
		prm = {on,off};//on means mute; off means unmute
		a*out* = {ao01};
		
		Description:
		Set Audio Output to mute/unmute.
			Command:
			SET MUTE *ao01* *on*<CR><LF>
			
			Return:
			MUTE *ao01* *on*<CR><LF>
			
			Description:
			Set Audio Output to mute/unmute.

No.	Description	Command	Example

4	Get Audio Output mute status.	**Command:**
		GET MUTE a*out*<CR><LF>
		
		Return:
		MUTE *aout pcm*<CR><LF>
		
		Parameter:
		pcm = {on,off};;//on means mute; off means unmute
		a*out* = {ao01};
		
		Description:
		Get Audio Output mute status.
		
		Default:
		off
			Command:
			GET MUTE *ao01*<CR><LF>
			
			Return:
			MUTE a*o01 off*<CR><LF>
			
			Description:
			Get Audio Output mute status.

No.	Description	Command	Example
		Scaler	

1	Set video Output scaler to on/off	**Command:**
		SET SCALER out prm<CR><LF>
		
		Return:
		SCALER out prm<CR><LF>
		
		Parameter:
		pcm = {on,off};//on means scaler; off means not scaler
		out = {o01, o02, o03, o04, all};
		
		Description:
		Set 4K-to-1080p simple scaler on each video output ON or OFF.
		out={o01=hdmiout1, o02=hdmiout2, o03=hdbtout1, o04=hdbtout2}
		all = {all};
			Command:
			SET SCALER o01 on<CR><LF>
			
			Return:
			SCALER o01 on<CR><LF>
			
			Description:
			Set hdmiout 1 scaler to on.

No.	Description	Command	Example

2	Get video Output scaler on/off status	**Command:**
		GET SCALER *out*<CR><LF>
		
		Return:
		SCALER *out* *prm*<CR><LF>
		
		Parameter:
		prm = {on, off};;//on means mute; off means unmute
		out = {o01, o02, o03, o04, all };
		
		Description:
		Get video Output scaler status.
		out={o01=hdmiout1, o02=hdmiout2, o03=hdbtout1, o04=hdbtout2}
		all = {all};
		Default:
		on
			Command:
			GET SCALER *o01*<CR><LF>
			
			Return:
			SCALER o01 *on*<CR><LF>
			
			Description:
			Get video Output 1 scaler status.

3	Volume gain increase	**Command:** SET VOLGAIN_INC a*out*<CR><LF> **Return:** VOLGAIN_INC a*out* *prm*<CR><LF> **Parameter:** a*out* = {ao01}; *prm*= {0~100} **Description:** Increase AUDIO OUT volume by step level. **Default:** 50	**Command:** SET VOLGAIN_INC a*o01*<CR><LF> **Return:** VOLGAIN_INC a*o01* *60*<CR><LF> **Description:** After increasing volume, AUDIO OUT volume becomes 60 of 100 in total.

No.	Description	Command	Example
4	Volume gain decrease	**Command:** SET VOLGAIN_DEC a*out* <CR><LF> **Return:** VOLGAIN_DEC a***out*** *prm*<CR><LF> **Parameter:** a*out* = {ao01}; *prm*= {0~100} **Description:** Decrease AUDIO OUT volume by step level. **Default:** 50	**Command:** SET VOLGAIN_DEC a*o01*<CR><LF> **Return:** VOLGAIN_DEC *ao01 40*<CR><LF> **Description:** After volume decreasing, the AUDIO OUT volume becomes 40.
5	Volume gain adjust	**Command:** SET VOLGAIN_DATA *aout prm*<CR><LF> **Return:** VOLGAIN_DATA *aout prm*<CR><LF> **Parameter:** *aout* = {ao01}; *prm* = {0~100} **Description:** VOLGAIN is volgain. **Default:** 50	**Command:** SET VOLGAIN_DATA *oa01 50*<CR><LF> **Return:** VOLGAIN_DATA *oa01 50*<CR><LF> **Description:** Set volume gain of audio output1 at 50.

No.	Description	Command	Example
6	Get current adjustment gain of volume	**Command:** GET VOLGAIN_DATA *aout*<CR><LF> **Return:** VOLGAIN_DATA *aout prm*<CR><LF> **Parameter:** *aout* = {ao01}; *prm* = {0~100} **Description:** VOLGAIN is volgain. {0~100} **Default:** 50	**Command:** GET VOLGAIN_DATA a*o01*<CR><LF> **Return:** VOLGAIN_DATA a*o01 50* <CR><LF> **Description:** Get volume gain, the gain is 50.
7	Set volume gain adjust step	**Command:** SET VOLGAIN_STEP *aout prm*<CR><LF> **Return:** VOLGAIN_STEP *aout prm*<CR><LF> **Parameter:** *aout* = {ao01}; *prm* = {1~10} **Default:** 5	**Command:** SET VOLGAIN_STEP *oa01 5*<CR><LF> **Return:** VOLGAIN_STEP *oa01 5*<CR><LF> **Description:** Set volume gain adjust step 5.

No.	Description	Command	Example
8	Set volume gain adjust step	**Command:** GET VOLGAIN_STEP *aout*<CR><LF> **Return:** VOLGAIN_STEP *aout prm*<CR><LF> **Parameter:** *aout* = {ao01}; *prm* = {1~10} **Default:** 5	**Command:** GET VOLGAIN_STEP a*o01*<CR><LF> **Return:** VOLGAIN_STEP a*o01 5* <CR><LF> **Description:** Get volume gain adjust step is 5.
USB Matrix			
1	Set USB Work Model	**Command:** SET USB_M *prm*<CR><LF> **Return:** USB_M *prm*<CR><LF> **Parameter:** *prm* = {follow, independent} **Description:** Set USB Work Model.	**Command:** SET USB_M *follow*<CR><LF> **Return:** USB_M *follow*<CR><LF> **Description:** Set usb work mode to follow.
2	Get USB Work Model	**Command:** GET USB_M<CR><LF> **Return:** USB_M *prm*<CR><LF> **Parameter:** *prm* = {follow, independent} **Description:** Get USB Work Model.	**Command:** GET USB_M<CR><LF> **Return:** USB_M *follow*<CR><LF> **Description:** Get usb work mode, and the work mode is follow.

No.	Description	Command	Example
13	Switch one USB source to one USB output	**Command:** SET USBSW *out in*<CR><LF> **Return:** USBSW *out in*<CR><LF> **Parameter:** *in* = {i00, i01, i02, i03, i04}; *out* = {o01, o02}; **Description:** i00 is blank. USBSW is short for usb switch. Set USB Switch from one USB input source to one USB output.	**Command:** SET USBSW *o01 i01*<CR><LF> **Return:** USBSW *o01 i01*<CR><LF> **Description:** Set USB Switch from USB input1 to USB output1
14	Switch one USB input to all outputs	**Command:** SET USBSW *all in*<CR><LF> **Return:** USBSW *all in*<CR><LF> **Parameter:** *in = {i00, i01, i02, i03, i04};* *out = {o01, o02};* *all* = {all}; **Description:** USBSW is short for usb switch Set USB Switch from one input source to all output sinks.	**Command:** SET USBSW *all i01*<CR><LF> **Return:** USBSW *all i01*<CR><LF> **Description:** Set USB Switch from in1 to all output sinks.

No.	Description	Command	Example
15	Get which usb input is mapping to the indicate Output	**Command:** GET USBSW *out*<CR><LF> **Return:** USBSW *out in*<CR><LF> **Parameter:** *in = {i00, i01, i02, i03, i04};* *out = {o01, o02};* **Description:** USBSW is short for usb switch. Get which usb input is mapping to the indicate Output.	**Command:** GET USBSW *o01*<CR><LF> **Return:** USBSW o01 *i02*<CR><LF> **Description:** Get which input is mapping to USB out1.
16	Get all correspondence among all USB inputs and outputs	**Command:** GET USBSW *all*<CR><LF> **Return:** USBSW *out in*<CR><LF> USBSW *out in*<CR><LF> **Parameter:** *in = {i00, i01, i02, i03, i04};* *out = {o01, o02};* *all*= {all}; **Description:** USBSW is short for usb switch. Get all correspondence among all USB inputs and outputs.	**Command:** GET USBSW *all* <CR><LF> **Return:** USBSW *o01 i01*<CR><LF> USBSW *o02 i02*<CR><LF> **Description:** Get all correspondence among all USB inputs and outputs.

No.	Description	Command	Example
4	Get USB Host Connection status	**Command:** GET USB_HOST_C in<CR><LF> **Return:** USB_HOST_C in *prm*<CR><LF> **Parameter:** in = {i01, i02, i03, i04} *prm* = {connect, unconnect} **Description:** Get USB Host Connection status.	**Command:** GET USB_HOST_C i01<CR><LF> **Return:** USB_HOST_C i01 *connect*<CR><LF> **Description:** Get USB Host Connection status.
Video Information			
1	Get video cable connection status	**Command:** GET VIDIN_CONNECT in<CR><LF> **Return:** VIDIN_CONNECT in *prm*<CR><LF> **Parameter:** *in= {i01, i02, i03, i04};* *prm* = {connect, unconnect} **Description:** Get input cable connection status.	**Command:** GET VIDIN_CONNECT i01<CR><LF> **Return:** VIDIN_CONNECT i01 connect<CR><LF> **Description:** Get i01 cable connection status.

No.	Description	Command	Example
2	Get output cable connection status	**Command:** GET VIDOUT_CONNECT OUT<CR><LF> **Return:** VIDOUT_CONNECT out *prm*<CR><LF> **Parameter:** out= *{o01, o02, o03, o04};* *prm* = {connect, unconnect} **Description:** Get output cable connection status.	**Command:** GET VIDOUT_CONNECT o01<CR><LF> **Return:** VIDOUT_CONNECT o01 connect<CR><LF> **Description:** Get o01 output connection status.
3	Get the status of the video input signal	**Command:** GET VIDIN_SIG *in*<CR><LF> **Return:** VIDIN_SIG *in prm*<CR><LF> **Parameter:** *in* = {i01, i02, i03, i04}; *prm* = {no, valid} **Description:** Get the status of the video input signal.	**Command:** GET VIDIN_SIG i01<CR><LF> **Return:** VIDIN_SIG i01 valid<CR><LF> **Description:** Get the status of the video input1

No.	Description	Command	Example
4	Get input video format information	**Command:** GET VIDIN_FORMAT *in*<CR><LF> **Return:** VIDIN_FORMAT i*n prm*<CR><LF> **Parameter:** in ={i01, i02, i03, i04}; 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 • 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 *i01 3840*x2160,60; none hdr;rgb;8bit<CR><LF> **Description:** Get the HDMI input1 video information

No.	Description	Command	Example
5	Get input audio format information	**Command:** GET AUDIN_FORMAT *in*<CR><LF> **Return:** AUDIN_FORMAT in *prm*<CR><LF> **Parameter:** in = {i01, i02, i03, i04}; prm = { } **Description:** Get the resolution of the video input. prm = {<NONE / PCM / COMPRESSED/ HBR>; <Sampling rate>}	**Command:** GET AUDIN_FORMAT *i01*<CR><LF> **Return:** AUDIN_FORMAT *i01 PCM*;48kHZ<CR><LF> **Description:** HDMI input1 supports PCM 48KHz.
6	Get hdcp version of input video	**Command:** GET VIDIN_HDCP *in*<CR><LF> **Return:** VIDIN_HDCP *in prm*<CR><LF> **Parameter:** in = {i01, i02, i03, i04}; prm = {no hdcp, hdcp1.4, hdcp2.2} **Description:** Get hdcp version of input video.	**Command:** GET VIDIN_HDCP *i01*<CR><LF> **Return:** VIDIN_HDCP *i01 no hdcp*<CR><LF> **Description:** Get the HDCP status of HDMI input1.

No.	Description	Command	Example
		HDBT UART	
1	Get hdbt uart property.	**Command:** GET UART_CFG out<CR><LF> **Return:** UART_CFG *out prm1 parm2 parm3 parm4*<CR><LF> **Parameter:** out = {5001, 5002 | UART1, UART2}; prm1 = {9600, 19200, 38400, 57600, 115200} prm2 = {NONE, ODD, EVEN} prm3 = {7,8} prm4 = {1,2} **Description:** Get hdbt uart property.	**Command:** GET UART_CFG *5001*<CR><LF> **Return:** UART_CFG *5001 9600 none 8 1*<CR><LF> **Description:** Get hdbt uart1 property.

No.	Description	Command	Example
2	Set hdbt uart property	**Command:** SET UART_CFG out *prm1 parm2 parm3 parm4*<CR><LF> **Return:** UART_CFG *out prm1 parm2 parm3 parm4*<CR><LF> **Parameter:** out = {5001,5002 | UART1, UART2}; prm1 = {9600, 19200, 38400, 57600, 115200} prm2 = {NONE, ODD, EVEN} prm3 = {7, 8} prm4 = {1, 2} **Description:** Set hdbt uart property.	**Command:** SET UART_CFG *5001 9600 none 8 1*<CR><LF> **Return:** UART_CFG *5001 9600 none 8 1*<CR><LF> **Description:** Set hdbt uart1 property.
3	Set serial port routing	**Command:** ROUTE out *prm1 parm2 parm3 parm4 DATA*<CR><LF> **Parameter:** out = {5001 | UART1}; prm1 = {9600, 19200, 38400, 57600,115200} prm2 = {NONE, ODD, EVEN} prm3 = {7,8} prm4 = {1,2} DATA = {hex data separated by Spaceshex, data need conversion into ASCII code} **Description:** Serial port routing.	**Command:** ROUTE 5001 9600 none 8 1 DATA<CR><LF> **Description:** Serial port routing

No.	Description	Command	Example
		API Function	
1	Set Input Connect API Function to On/Off	**Command:** SET VIDIN_CONNECT_REPORT in *prm0* <CR><LF> **Return:** VIDIN_CONNECT_REPORT *in prm0* <CR><LF> **Parameters:** in = {i01, i02, i03, i04, all}; prm0 = {on, off}; **Description:** FN is short for function When the prm is *on,* input connect API Function is in active status. When the prm is *off*, input connect API Function is in inactive status.	**Command:** SET VIDIN_CONNECT_REPORT i01 *on* <CR><LF> **Return:** VIDIN_CONNECT_REPORT i01 *on* <CR><LF> **Description:** Set input connect API Function to active status.
2	Get Input Connect API Function On/Off status	**Command:** GET VIDIN_CONNECT_REPORT in<CR><LF> **Return:** VIDIN_CONNECT_REPORT *prm1 prm0*<CR><LF> **Parameters:** in = {i01, i02, i03, i04, all }; prm0= {on, off} prm1 = {i01, i02, i03, i04}; **Description:** Get input connect API Function is in active status or inactive status.	**Command:** GET VIDIN_CONNECT_REPORT i01 <CR><LF> **Return:** VIDIN_CONNECT_REPORT i01 *on* <CR><LF> **Description:** Get input connect API Function is in active status or inactive status. The result is in active status.

No.	Description	Command	Example
New Function			
1	Set the hostname of the device	**Command:** SET HOSTNAME prm<CR><LF> **Return:** HOSTNAME prm<CR><LF> **Parameter:** *prm* = {ascii characters of length less than or equal to 31} **Description:** Set the hostname of the device.	**Command:** SET HOSTNAME 123456789<CR><LF> **Return:** HOSTNAME 123456789<CR><LF> **Description:** Set the device's hostname to 123456789.
2	Get the device's hostname	**Command:** GET HOSTNAME<CR><LF> **Return:** HOSTNAME prm<CR><LF> **Parameter:** *prm* = {ascii characters of length less than or equal to 31} **Description:** Get the device's hostname.	**Command:** GET HOSTNAME<CR><LF> **Return:** HOSTNAME 123456789<CR><LF> **Description:** The device hostname is 123456789.

No.	Description	Command	Example
3	Get the device's serial number	**Command:** GET SN<CR><LF> **Return:** SN prm<CR><LF> **Parameter:** *prm* = {ascii characters of length less than or equal to 31} **Description:** Get the device's serial number.	**Command:** GET SN<CR><LF> **Return:** SN 123456789<CR><LF> **Description:** The device serial number is 123456789.
4	Get device's power status	**Command:** GET DEVICE_PWR<CR><LF> **Return:** DEVICE_PWR prm<CR><LF> **Parameter:** *prm* = {ON, OFF} **Description:** Returns ON if the device's power supply is normal, otherwise returns OFF.	**Command:** GET DEVICE_PWR<CR><LF> **Return:** DEVICE_PWR ON<CR><LF> **Description:** Current power supply to the device is normal.

No.	Description	Command	Example
5	Get the HDBT link status of the device	**Command:** GET HDBT_LINK in<CR><LF> **Return:** HDBT_LINK in prm<CR><LF> **Parameter:** in = {i01, i02, all}; *prm* = {CONNECTED, DISCONNECTED} **Description:** Get the HDBT link status of the device.	**Command:** GET HDBT_LINK i01<CR><LF> **Return:** HDBT_LINK i01 CONNECTED<CR><LF> **Description:** Currently, HDBT 1 port is connected.

syscomtec Distribution AG
Raiffeisenallee 8
D-82041 Oberhaching (bei München)
Tel.: +49 89 666 109 330
Email: post@syscomtec.com
https://www.syscomtec.com