

RS232 Default Setting

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

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.
<CR><LF> denotes a carriage return or a line feed; all commands must be ended up with a carriage return or a line feed.

| IDX | Description | Command | Example |
|---------------------------|--|--|--|
| Normal switch case | | | |
| 1 | Switch Input for Output | <p>Command: SET SW <i>in out</i><CR><LF></p> <p>Return: SW <i>in out</i><CR><LF></p> <p>Parameter: <i>in</i> = {in1, in2, in3, in4}; <i>out</i> = {out1,out2};</p> <p>Description: SW is short for Switch Switch one input source for one output sink</p> | <p>Command: SET SW in1 out2<CR><LF></p> <p>Return: SW in1 out2<CR><LF></p> <p>Description: Switch input 1 for hdmi output 2</p> |
| 2 | Get which input mapping to the indicate Output | <p>Command: GET MP <i>out</i><CR><LF></p> <p>Return: Mp <i>in out</i><CR><LF></p> <p>Parameter: <i>in</i> = {in1, in2, in3, in4}; <i>out</i> = {out1,out2};</p> <p>Description: MP is short for mapping Get which input mapping to the indicate Output</p> | <p>Command: GET MP out1<CR><LF></p> <p>Return: MP in1 out1<CR><LF></p> <p>Description: Get which input mapping to output 1</p> |
| 3 | Switch indicate input for all outputs | <p>Command: SET SW <i>in all</i><CR><LF></p> <p>Return: SW <i>in all</i> <CR><LF></p> <p>Parameter: <i>in</i> = {in1, in2, in3, in4}; <i>all</i> = {all};</p> <p>Description: SW is short for Switch Switch one input source for all output sink</p> | <p>Command: SET SW in1 all <CR><LF></p> <p>Return: SW in1 all<CR><LF></p> <p>Description: Switch input1 for all output sink</p> |

| IDX | Description | Command | Example |
|--------------------|---------------------------------------|---|---|
| 4 | Get which output mapping to all input | <p>Command: GET MP all<CR><LF></p> <p>Return: MP in out<CR><LF> MP in out<CR><LF></p> <p>Parameter: in = {in1, in2, in3, in4}; all = {all};</p> <p>Description: MP is short for mapping Get which output mapping to all input</p> | <p>Command: GET MP all <CR><LF></p> <p>Return: MP in1 out1<CR><LF> MP in2 out2<CR><LF></p> <p>Description: Get which output mapping to all input</p> |
| CEC Control | | | |
| 5 | Set CEC POWER ON/OFF | <p>Command: SET CEC_PWR <i>out prm</i><CR><LF></p> <p>Return: CEC_PWR <i>out prm</i><CR><LF></p> <p>Parameter: <i>prm</i> = {on, off} <i>out</i> = {out1, out2};</p> <p>Description: Set sink power on or off</p> | <p>Command: SET CEC_PWR out1 <i>on</i><CR><LF></p> <p>Return: CEC_PWR out1 <i>on</i><CR><LF></p> <p>Description: Set sink hdmi output 1 power on</p> |
| 6 | Set CEC AUTO POWER ON/OFF | <p>Command: SET AUTOCEC_FN <i>out prm</i><CR><LF></p> <p>Return: AUTOCEC_FN <i>out prm</i><CR><LF></p> <p>Parameter: <i>prm</i> = {on, off} <i>out</i> = {out1, out2};</p> <p>Description: Set sink auto power Function ON or OFF</p> | <p>Command: SET AUTOCEC_FN out1 <i>on</i><CR><LF></p> <p>Return: AUTOCEC_FN out1 <i>on</i><CR><LF></p> <p>Description: Set sink hdmi output 1 auto power ON</p> |

| IDX | Description | Command | Example |
|-----|--|---|---|
| 7 | Get CEC AUTO POWER ON/OFF Status | <p>Command: GET AUTOCEC_FN <i>out</i><CR><LF></p> <p>Return: AUTOCEC_FN <i>out</i> <i>prm</i><CR><LF></p> <p>Parameter: <i>prm</i> = {on, off} <i>out</i> = {out1, out2};</p> <p>Description: Get Sink auto power Function ON or OFF Status.</p> | <p>Command: GET AUTOCEC_FN out1<CR><LF></p> <p>Return: AUTOCEC_FN <i>out1</i> <i>on</i></p> <p>Description: Get Sink auto power status, and the status is ON.</p> |
| 8 | Set CEC POWER Delay Time | <p>Command: SET AUTOCEC_D <i>out</i> <i>prm</i><CR><LF></p> <p>Return: AUTOCEC_D <i>out</i> <i>prm</i><CR><LF></p> <p>Parameter: <i>out</i> = {out1,out2}; <i>prm</i> = {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.</p> <p>Description: AUTOCEC_D is short for CEC auto Power Delay Timing</p> | <p>Command: SET AUTOCEC_D out1 2<CR><LF></p> <p>Return: AUTOCEC_D out1 2<CR><LF></p> <p>Description: when no active signal to hdmiout1, 2 minutes later, the unit will auto power off.</p> |

| IDX | Description | Command | Example |
|-----|---------------------------------------|---|--|
| 9 | Get CEC POWER Delay Time Status | <p>Command: GET AUTOCEC_D <i>out</i> <CR><LF></p> <p>Return: AUTOCEC_D <i>out</i> <i>prm</i><CR><LF></p> <p>Parameter: <i>out</i> = {out1, out2}; <i>prm</i> = {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.</p> <p>Description: AUTOCEC_D is short for CEC auto Power Delay Timing</p> | <p>Command: GET AUTOCEC_D out1 <CR><LF></p> <p>Return: AUTOCEC_D <i>out1 2</i> <CR><LF></p> <p>Description: Get hdmiout1 auto power delay time, the result is 2 minutes</p> |

| IDX | Description | Command | Example |
|-------------|----------------|---|--|
| EDID | | | |
| 10 | Set Input EDID | <p>Command: SET EDID <i>in prm</i><CR><LF></p> <p>Return: EDID <i>in prm</i><CR><LF></p> <p>Parameter: <i>in</i> = {in1, in2, in3, in4}; <i>prm</i> = {1 ~ 11}</p> <p>1) Restore Defaults; 2) Copy EDID from HDMI out1; 3) Copy EDID from HDMI out2; 4) 4K@60Hz 4:4:4, 2.0ch, with HDR; 5) 4K@60Hz 4:4:4, 5.1ch, with HDR; 6) 4K@60Hz 4:4:4, 7.1ch, with HDR; 7) 4K@30Hz 4:4:4, 2.0ch, with HDR and 4:2:0; 8) 4K@30Hz 4:4:4, 7.1ch, with HDR and 4:2:0; 9) 1080P@60Hz, 2.0ch; 10) 1080P@60Hz, 5.1ch; 11) 1080P@60Hz, 7.1ch ;</p> <p>Description: Set Input EDID</p> | <p>Command: SET EDID <i>in1 4</i><CR><LF></p> <p>Return: EDID <i>in1 4</i><CR><LF></p> <p>Description: Set in1 EDID 4K@60Hz 4:4:4, 2.0ch, with HDR;</p> |

| IDX | Description | Command | Example |
|-----|---------------------------|---|--|
| 11 | Get All Input EDID status | <p>Command: GET EDID <i>all</i> <CR><LF></p> <p>Return: EDID <i>in prm</i><CR> EDID <i>in prm</i><CR> EDID <i>in prm</i><CR><LF></p> <p>Parameter: in = {in1,in2,in3,in4}; prm = {1 ~11} 1) Restore Defaults; 2) Copy EDID from HDMI out1; 3) Copy EDID from HDMI out2; 4) 4K@60Hz 4:4:4, 2.0ch, with HDR; 5) 4K@60Hz 4:4:4, 5.1ch, with HDR; 6) 4K@60Hz 4:4:4, 7.1ch, with HDR; 7) 4K@30Hz 4:4:4, 2.0ch, with HDR and 4:2:0; 8) 4K@30Hz 4:4:4, 7.1ch, with HDR and 4:2:0; 9) 1080P@60Hz, 2.0ch; 10) 1080P@60Hz, 5.1ch; 11) 1080P@60Hz, 7.1ch ;</p> <p>Description: Get all input EDID Status</p> | <p>Command: GET EDID <i>all</i> <CR><LF></p> <p>Return: EDID in1 <i>1</i><CR> EDID in2 <i>2</i><CR></p> <p>Description: Get all input EDID Status</p> |

| IDX | Description | Command | Example |
|--------------------|---------------------------|--|--|
| 12 | Get one input EDID Status | <p>Command: GET EDID <i>in</i> <CR><LF></p> <p>Return: EDID <i>in prm</i><CR><LF></p> <p>Parameter: <i>in</i> = {in1, in2, in3, in4}; <i>prm</i> = {1 ~11}</p> <p>1) Restore Defaults; 2) Copy EDID from HDMI out1; 3) Copy EDID from HDMI out2; 4) 4K@60Hz 4:4:4, 2.0ch, with HDR; 5) 4K@60Hz 4:4:4, 5.1ch, with HDR; 6) 4K@60Hz 4:4:4, 7.1ch, with HDR; 7) 4K@30Hz 4:4:4, 2.0ch, with HDR and 4:2:0; 8) 4K@30Hz 4:4:4, 7.1ch, with HDR and 4:2:0; 9) 1080P@60Hz, 2.0ch; 10) 1080P@60Hz, 5.1ch; 11) 1080P@60Hz, 7.1ch ;</p> <p>Description: Get one input EDID Status</p> | <p>Command: GET EDID <i>in1</i><CR><LF></p> <p>Return: EDID <i>in1</i> <i>4</i><CR><LF></p> <p>Description: Get <i>in1</i> edid status, and the status is 4K@60Hz 4:4:4, 2.0ch, with HDR;</p> |
| System Info | | | |
| 13 | Factory reset | <p>Command: RESET<CR><LF></p> <p>Return: RESET<CR><LF></p> <p>Description: Factory reset</p> | <p>Command: RESET<CR><LF></p> <p>Return: RESET<CR><LF></p> <p>Description: Factory reset all board</p> |

| IDX | Description | Command | Example |
|-----|--------------------------------------|---|---|
| 14 | System reboot | Command: REBOOT<CR><LF> Return: REBOOT<CR><LF> Description: System reboot | Command: REBOOT<CR><LF> Return: REBOOT<CR><LF> Description: System reboot |
| 15 | Get selected target firmware version | Command: GET VER<CR><LF> Return: VER <i>prm</i> <CR><LF> Parameter: <i>prm</i> = {...} // according to actual firmware version Description: Get selected target firmware version | Command: GET VER<CR><LF> Return: VER 1.0<CR><LF> Description: Get firmware version |
| 16 | Set IR System Code | Command: Set IR_SC <i>prm</i> <CR><LF> Return: IR_SC <i>prm</i> <CR><LF> Parameter: <i>prm</i> = { <i>all</i> , <i>mode1</i> , <i>mode2</i> }; <i>mode1</i> = 0x00 <i>mode2</i> = 0x4e Description: Set IR System Code | Command: Set IR_SC <i>mode1</i> <CR><LF> Return: IR_SC <i>mode1</i> <CR><LF> Description: Set IR System code mode 1 |

| IDX | Description | Command | Example |
|-------------------|-----------------------|---|---|
| 17 | Get IR System Code | <p>Command: Get IR_SC <CR><LF></p> <p>Return: IR_SC <i>prm</i><CR><LF></p> <p>Parameter: <i>prm</i> = {all, mode1, mode2}; <i>mode1</i> = 0x00 <i>mode2</i> = 0x4e</p> <p>Description: Get IR System Code</p> | <p>Command: Get IR_SC <CR><LF></p> <p>Return: IR_SC <i>mode1</i><CR><LF></p> <p>Description: Get IR System code, IR System code is mode 1</p> |
| 18 | Get the API list | <p>Command: help<CR><LF></p> <p>Description: Get the API list</p> | <p>Command: help<CR><LF></p> <p>Description: Get the API list</p> |
| Audio Mute | | | |
| 19 | Set Audio Output mute | <p>Command: SET MUTE <i>out</i> <i>pcm</i><CR><LF></p> <p>Return: MUTE <i>out</i> <i>pcm</i><CR><LF></p> <p>Parameter: <i>pcm</i> = {on, off}; //on means mute; off means unmute <i>out</i> = {hdmaudioout1, hdmaudioout2, spdifaudioout1, spdifaudioout2, audioout1, audioout2};</p> <p>Description: Set Audio mute or not mute.</p> | <p>Command: SET MUTE <i>audioout1</i> <i>on</i><CR><LF></p> <p>Return: MUTE <i>audioout1</i> <i>on</i><CR><LF></p> <p>Description: Set audioout1 mute on</p> |

| IDX | Description | Command | Example |
|-----|------------------------------|---|---|
| 20 | Get Audio Output mute status | <p>Command: GET MUTE <i>out</i><CR><LF></p> <p>Return: MUTE <i>out pcm</i><CR><LF></p> <p>Parameter: pcm = {on, off};; //on means mute; off means unmute <i>out</i> = {hdmiaudioout1, hdmiaudioout2, spdifaudioout1, spdifaudioout2, audioout1, audioout2, all};</p> <p>Description: Get Audio Output mute status</p> | <p>Command: GET MUTE <i>audioout1</i><CR><LF></p> <p>Return: MUTE <i>audioout1 pcm</i><CR><LF></p> <p>Description: Get Audio Output mute status.</p> |