

MX988VM RS232 ASCII Protocol

Version	V1.0
Date	2020-09-07
Draft by	xll

Content

MX988VM RS232 ASCII Protocol.....	0
Serial communication protocol format.....	2
1 Input Board Command.....	3
1.1 Video routing.....	3
1.2 Input Type.....	3
1.3 Input Signal format.....	5
1.4 Output Type.....	5
1.5 Output Signal format.....	6
1.6 Software Version.....	6
1.7 Write Input Edid.....	7
1.8 Input Board Audio Select.....	8
2 Output Board Command.....	9
2.1 Input Signal format.....	9
2.2 Output Type.....	9
2.3 Output Signal format.....	10
2.4 Brightness Setting.....	12
2.5 Contrast Setting.....	12
2.6 Saturation Setting.....	13
2.7 Sharpness Setting.....	13
2.8 Picture Quality Reset.....	14
2.9 Software Version.....	14
2.10 Set TV-WALL.....	15
2.11 Read EDID.....	16
3. System command.....	17
3.1 Device size.....	17
3.2 Device IP.....	17
3.3 System Reset.....	17
3.4 Panel Lock.....	18
3.5 System Version.....	18
3.6 TVWALL Mode.....	19

Serial communication protocol format

Baud Rate: 9600

Data bits: 8

Parity: None

Stop bits: 1

Operation (3 byte)	Spacer (1 byte)	Target (N bytes)	Spacer (1 byte)	Command type (N bytes)	Command parameters (N bytes)	Command tail (1 byte)
SET/GET	Space	The target that handles this command.	Space	Command type	[Parameter1] [Parameter2]	↵ This is ASCII carriage return 0x0d

Notes:

Space is the ASCII character 0x20

↵ Represents the ASCII character 0x0d

All Return messages are always terminated by CR/LF, the ASCII characters 0x0d 0x0a

All items shown in square brackets, [], are optional.

Any SET command that contains leading zeroes should not include the leading zeros in any response message.

The value ranges for certain commands are not given, please state and minimum and maximum values for each command that uses a numerical value range.

1 Input Board Command

1.1 Video routing

Operation (3B)	Spacer (1B)	Target (3B/4B/5B)	Spacer (1B)	Command (5B)	Command parameters (4B/5B/6B)	Command tail (1B)
SET	Space	INx/INxx/INxxx x is the input port number	Space	VIDEO	OUTa/OUTaa/OUTaaa or ALL	↵ This is ASCII carriage return 0x0d

A. Set video route: Input port-x/xx/xxx switch to output port-a/aa/aaa , or all output ports

For example, SET video route: Input port 1 switch to output port 1

Send: SET IN1 VIDEO OUT1↵

Receive: IN1 VIDEO OUT1

For example, SET video route: Input port 1 switch to all output ports

Send: SET IN1 VIDEO ALL↵

Receive: IN1 VIDEO ALL

1.2 Input Type

Operation (3B)	Spacer (1B)	Target (3B/4B/5B)	Spacer (1B)	Command (7B)	Command tail (1B)
GET	Space	INx/INxx/INxxx x is the input port number For example, IN1/IN01/IN001	Space	IN-TYPE	↵ This is ASCII carriage return 0x0d

A. GET input type of input board:

Send: GET IN1 IN-TYPE↵

Receive: IN1 IN-TYPE UHD

Send: GET IN2 IN-TYPE↵

Receive: IN1 IN-TYPE HDMI-BYPASS

Send: GET IN3 IN-TYPE↵

Receive: IN3 IN-TYPE DVI

Send: GET IN5 IN-TYPE↵

Receive: IN5 IN-TYPE HDBST

Send: GET IN7 IN-TYPE↵

Receive: IN7 IN-TYPE SDI

NOTE:

1. Total input type list is as followed:

FIBER, SDI, HDBST, DVI, CVBS, YUV, VGA, HDMI, HDMI-BYPASS,
UHD

2. Read only input type list is as followed:

FIBER, SDI, HDBST, HDMI-BYPASS

3. The specific setting type depends on the type of device
4. Some board only support read, not support set;

1.3 Input Signal format

Operation (3B)	Spacer (1B)	Target (3B/4B/5Bs)	Spacer (1B)	Command (9B)	Command parameters (0 B)	Command tail (1B)
GET	Space	INx/INxx/INxxx x is the input port number	Space	IN-SIGNAL	Send: Null (0B) Receive: TYPE@Resolution (N bytes) or TYPE@NO-SIGNAL	↵ This is ASCII carriage return 0x0d

A. GET input signal format of input board:

Send: GET IN1 IN-SIGNAL↵

Receive: IN1 IN-SIGNAL UHD@3840x2160p30

Send: GET IN2 IN-SIGNAL↵

Receive: IN1 IN-SIGNAL HDMI-BYPASS@3840x2160p30

Send: GET IN3 IN-SIGNAL↵

Receive: IN3 IN-SIGNAL DVI@1920x1080p60

Send: GET IN5 IN-SIGNAL↵

Receive: IN5 IN-SIGNAL HDBST@1920x1080p60

Send: GET IN7 IN-SIGNAL↵

Receive: IN7 IN-SIGNAL SDI@1280x0720p60

NOTE:

1. Total input type list is as follows:

FIBER, SDI, HDBST,
DVI, CVBS, YUV, VGA, HDMI,
HDMI-BYPASS, UHD

1.4 Output Type

Operation (3B)	Spacer (1B)	Target (3B/4B/5B)	Spacer (1B)	Command (8B)	Command parameters (0B or 4B/3B)	Command tail (1B)
GET	Space	INx/INxx/INxxx x is the input port number	Space	OUT-TYPE	Send:Null (0B) or Receive: HDMI/DVI	↵ This is ASCII carriage return 0x0d

A. GET output type of input board:

Send: GET IN1 OUT-TYPE↵

Receive: IN1 OUT-TYPE HDMI

NOTE:

1. The output type of input board only support ~~DVI~~ and HDMI
2. SET not support

Operation (3B)	Spacer (1B)	Target (3B/4B/5B)	Spacer (1B)	Command (7B)	Command tail (1B)
GET	Space	INx/INxx/INxxx x is the input port number	Space	VERSION	↵ This is ASCII carriage return 0x0d

Output Signal format

Operation (3B)	Spacer (1B)	Target (3B/4B/5B)	Spacer (1B)	Command (10B)	Command parameters (0B or NB)	Command tail (1B)
GET	Space	INx/INxx/INxxx x is the input port number	Space	OUT-SIGNAL	Send:Null (0B) or Receive:TYPE@Resolution (NB)	↵ This is ASCII carriage return 0x0d

A. GET output signal format of input board:

Send: GET IN1 OUT-SIGNAL ↵

Receive: IN1 OUT-SIGNAL HDMI@3840x2160p30

Send: GET IN3 OUT-SIGNAL ↵

Receive: IN3 OUT-SIGNAL HDMI@1920x1080p60

Send: GET IN5 OUT-SIGNAL ↵

Receive: IN5 OUT-SIGNAL HDMI@1920x1080p60

NOTE:

- The output type of input board only support DVI and HDMI
- Some input board's output resolution cannot be changed, such as HDMI-BYPASS
- The supported output resolution is depend on the input board's type
 - DVI-U input board support output resolution is as followed:

“1024x768p60”, “1280x720p60”, “1280x1024p60”, “1600x1200p60”,
 “1680x1050p60”, “1920x1080p50”, “1920x1080p60”,
 - UHD input board support output resolution is as followed:

“1280x720p50”, “1280x720p60”, “1920x1080p50”, “1920x1080p60”, “4K2Kp25”,
 “4K”, “4K2Kp50”, “4K2Kp60”, “1024x768p60”, “1280x768p60”,
 “1280x1024p60”, “1360x768p60”, “1366x768p60”, “1400x1050p60”, “1600x1200p60”,
 “1920x1200p60”,
- SMS88-18G only support GET, not support SET

1.6 Software Version

Get the software version of input board:

Send: GET IN1 VERSION↵

Receive: GET IN1 VERSION 2019/01/01-12:00:00

Send: GET IN01 VERSION↵

Receive: GET IN01 VERSION 2019/01/01-12:00:00

Send: GET IN001 VERSION↵

Receive: GET IN001 VERSION 2019/01/01-12:00:00

1.7 Write Input Edid

Operation (3B)	Spacer (1B)	Target (3B/4B/5B)	Spacer (1B)	Command (4B)	Command parameters (53B/54B)	Command tail (1B)
SET	Space	INx/INxx/INxxx x is the input port number	Space	EDID	PART1 d1 d2 ... d16 PART2 d1 d2 ... d16 PART16 d1 d2 ... d16	↵ This is ASCII carriage return 0x0d

SET (Write) EDID data to input port

For example, SET (Write) EDID data to input port 1 (Write 16 times)

Send: SET IN1 EDID PART1 00 FF FF FF FF FF FF 00 63 18 22 00 66 00 00 00↵

Receive: IN1 EDID PART1 00 FF FF FF FF FF FF 00 63 18 22 00 66 00 00 00

Send: SET IN1 EDID PART2 05 1C 01 03 80 59 32 78 0A EE 91 A3 54 4C 99 26↵

Receive: IN1 EDID PART2 05 1C 01 03 80 59 32 78 0A EE 91 A3 54 4C 99 26

Send: SET IN1 EDID PART3 0F 50 54 01 08 00 81 C0 81 C0 81 00 81 80 95 00↵

Receive: IN1 EDID PART3 0F 50 54 01 08 00 81 C0 81 C0 81 00 81 80 95 00

Send: SET IN1 EDID PART4 A9 C0 B3 00 01 01 08 E8 00 30 F2 70 5A 80 B0 58↵

Receive: IN1 EDID PART4 A9 C0 B3 00 01 01 08 E8 00 30 F2 70 5A 80 B0 58

Send: SET IN1 EDID PART5 8A 00 C4 8E 21 00 00 1E 02 3A 80 18 71 38 2D 40↵

Receive: IN1 EDID PART5 8A 00 C4 8E 21 00 00 1E 02 3A 80 18 71 38 2D 40

Send: SET IN1 EDID PART6 58 2C 45 00 50 1D 74 00 00 1E 00 00 00 FD 00 17↵

Receive: IN1 EDID PART6 58 2C 45 00 50 1D 74 00 00 1E 00 00 00 FD 00 17

Send: SET IN1 EDID PART7 3C 0F 88 3C 00 0A 20 20 20 20 20 20 00 00 00 FC↵

Receive: IN1 EDID PART7 3C 0F 88 3C 00 0A 20 20 20 20 20 20 00 00 00 FC

Send: SET IN1 EDID PART8 00 48 44 4D 49 0A 20 20 20 20 20 20 20 01 16↵

Receive: IN1 EDID PART8 00 48 44 4D 49 0A 20 20 20 20 20 20 20 01 16

Send: SET IN1 EDID PART9 02 03 40 F1 55 61 10 1F 04 13 05 14 20 21 22 5D↵

Receive: IN1 EDID PART9 02 03 40 F1 55 61 10 1F 04 13 05 14 20 21 22 5D

Send: SET IN1 EDID PART10 5E 5F 60 65 66 07 12 03 16 01 23 09 07 07 83 01↵

Receive: IN1 EDID PART10 5E 5F 60 65 66 07 12 03 16 01 23 09 07 07 83 01

Send: SET IN1 EDID PART11 00 00 6E 03 0C 00 30 00 B8 3C 21 10 80 01 02 03↵

Receive: IN1 EDID PART11 00 00 6E 03 0C 00 30 00 B8 3C 21 10 80 01 02 03

Send: SET IN1 EDID PART12 04 67 D8 5D C4 01 78 80 03 E2 00 4F E3 0F 01 E0↵

Receive: IN1 EDID PART12 04 67 D8 5D C4 01 78 80 03 E2 00 4F E3 0F 01 E0

Send: SET IN1 EDID PART13 01 1D 80 D0 72 1C 16 20 10 2C 25 80 50 1D 74 00↵

Receive: IN1 EDID PART13 01 1D 80 D0 72 1C 16 20 10 2C 25 80 50 1D 74 00

Send: SET IN1 EDID PART14 00 9E 66 21 56 AA 51 00 1E 30 46 8F 33 00 50 1D↵

Receive: IN1 EDID PART14 00 9E 66 21 56 AA 51 00 1E 30 46 8F 33 00 50 1D

Send: SET IN1 EDID PART15 74 00 00 1E 00 00 00 00 00 00 00 00 00 00 00↵

Receive: IN1 EDID PART15 74 00 00 1E 00 00 00 00 00 00 00 00 00 00 00

Send: SET IN1 EDID PART16 00 00 00 00 00 00 00 00 00 00 00 00 00 00 A1↵

Receive: IN1 EDID PART16 00 00 00 00 00 00 00 00 00 00 00 00 00 00 A1

NOTE:

1. EDID totally have 256 bytes, so we need to write 16 times and 16 bytes will be write per time.

1.8 Input Board Audio Select

Operation (3B)	Spacer (1B)	Target (3B/4B/5B)	Spacer (1B)	Command (5B)	Command tail (1B)
GET	Space	INx/INxx/INxxx x is the input port number	Space	AUDIO	↵ This is ASCII carriage return 0x0d

For example:

Send: GET IN1 AUDIO↵

Receive: IN1 AUDIO EMBEDDED

Send: GET IN1 AUDIO↵

Receive: IN1 AUDIO L/R

Operation (3B)	Spacer (1B)	Target (3B/4B/5B)	Spacer (1B)	Command (5B)	Command parameters (NB)	Command tail (1B)
GET	Space	INx/INxx/INxxx x is the input port number	Space	AUDIO	L/R or EMBEDDED	↵ This is ASCII carriage return 0x0d

For example:

Send: SET IN1 AUDIO L/R

Receive: IN1 AUDIO L/R

Send: SET IN1 AUDIO EMBEDDED

Receive: IN1 AUDIO EMBEDDED

2 Output Board Command

2.1 Input Signal format

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (9B)	Command parameters (0B/NB)	Command tail (1B)
GET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	IN-SIGNAL	Send: Null Receive: TYPE@Resolution	↵ This is ASCII carriage return 0x0d

GET input signal format of output board:

Send: GET OUT1 IN-SIGNAL↵

Receive: OUT1 IN-SIGNAL UHD-HDMI@4K2Kp30

Send: GET OUT3 IN-SIGNAL↵

Receive: OUT3 IN-SIGNAL HDMI@1920x1080p60

Send: GET OUT5 IN-SIGNAL↵

Receive: OUT5 IN-SIGNAL HDMI@1920x1080p60

Send: GET OUT7 IN-SIGNAL↵

Receive: OUT7 IN-SIGNAL HDMI@1920x1080p60

NOTE:

1. Input type of output board only support DVI , HDMI and UHD-HDMI

2.2 Output Type

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (8B)	Command tail (1B)
GET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	OUT-TYPE	↵ This is ASCII carriage return 0x0d

GET output type of output board:

Send: GET OUT1 OUT-TYPE↵

Receive: OUT1 OUT-TYPE UHD-HDMI

Send: GET OUT3 OUT-TYPE↵

Receive: OUT3 OUT-TYPE HDMI

Send: GET OUT5 OUT-TYPE↵

Receive: OUT5 OUT-TYPE HDBST

Send: GET OUT7 OUT-TYPE↵

Receive: OUT7 OUT-TYPE SDI

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (8B)	Command parameters (NB)
SET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	OUT-TYPE	CVBS/VGA/YUV/HDMI/DVI

SET output type of output board:

Send: SET OUT1 OUT-TYPE UHD-HDMI↵

Receive: OUT1 OUT-TYPE UHD-HDMI
 Send: SET OUT3 OUT-TYPE DVI↵
 Receive: OUT01 OUT-TYPE DVI
 Send: SET OUT3 OUT-TYPE CVBS↵
 Receive: OUT3 OUT-TYPE CVBS
 Send: SET OUT3 OUT-TYPE YUV↵
 Receive: OUT3 OUT-TYPE YUV
 Send: SET OUT3 OUT-TYPE VGA↵
 Receive: OUT3 OUT-TYPE VGA
 Send: SET OUT3 OUT-TYPE HDMI↵
 Receive: OUT3 OUT-TYPE HDMI

NOTE:

1. Total output type list is as followed:
 FIBER, SDI, HDBST, HDBST-HDMI, HDBST-DVI, DVI, CVBS, YUV, VGA, HDMI, HDMI-BYPASS, HDMI-BYPASS-HDCP-ON, HDMI-BYPASS-HDCP-OFF, UHD-HDMI, UHD-HDMI-1.4, UHD-HDMI-2.2, UHD-DVI
2. Read only output type list is as followed:
 FIBER, SDI, HDBST, HDMI-BYPASS
3. Different board support different parameter;
4. SMS88-18G support UHD-HDMI/UHD-DVI/UHD-HDMI-1.4/UHD-HDMI-2.2;

2.3 Output Signal format

Operation type (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (10B)	Command tail (1B)
GET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	OUT-SIGNAL	↵ This is ASCII carriage

GET output signal format of output board:

Send: GET OUT1 OUT-SIGNAL↵
 Receive: OUT1 OUT-SIGNAL UHD-HDMI@4K2Kp60
 Send: GET OUT3 OUT-SIGNAL↵
 Receive: OUT3 OUT-SIGNAL HDMI@1920x1080p60
 Send: GET OUT5 OUT-SIGNAL↵
 Receive: OUT5 OUT-SIGNAL HDBST@1920x1080p60
 Send: GET OUT7 OUT-SIGNAL↵
 Receive: OUT7 OUT-SIGNAL SDI@1280x720p60

Operation type (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (10B)	Command parameters (NB)
SET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	OUT-SIGNAL	TYPE@Resolution

SET output signal format of output board:

Send: SET OUT1 OUT-SIGNAL 1024x768p60↵
 Receive: OUT1 OUT-SIGNAL HDMI@1024x768p60
 Send: SET OUT3 OUT-SIGNAL 1024x768p60↵

Receive: OUT01 OUT-SIGNAL HDMI@1024x768p60
 Send: SET OUT3 OUT-SIGNAL 1280x720p60↵
 Receive: OUT3 OUT-SIGNAL HDMI@1280x720p60
 Send: SET OUT3 OUT-SIGNAL 1280x1024p60↵
 Receive: OUT3 OUT-SIGNAL HDMI@1280x1024p60
 Send: SET OUT3 OUT-SIGNAL 1360x768p60↵
 Receive: OUT3 OUT-SIGNAL HDMI@1360x768p60
 Send: SET OUT3 OUT-SIGNAL 1600x1200p60↵
 Receive: OUT3 OUT-SIGNAL HDMI@1600x1200p60
 Send: SET OUT3 OUT-SIGNAL 1680x1050p60↵
 Receive: OUT3 OUT-SIGNAL HDMI@1680x1050p60
 Send: SET OUT3 OUT-SIGNAL 1920x1080p30↵
 Receive: OUT3 OUT-SIGNAL HDMI@1920x1080p30
 Send: SET OUT3 OUT-SIGNAL 1920x1080p60↵
 Receive: OUT3 OUT-SIGNAL HDMI@1920x1080p60

NOTE:

- 1 Total output type list is as followed:
 FIBER, SDI,
 HDBST, HDBST-HDMI, HDBST-DVI,
 DVI, CVBS, YUV, VGA, HDMI,
 HDMI-BYPASS, HDMI-BYPASS-HDCP-ON, HDMI-BYPASS-HDCP-OFF
 UHD-HDMI, UHD-1.4, UHD-2.2, UHD-DVI
- 2 Read only output type list is as followed:
 FIBER, SDI, HDBST, HDMI-BYPASS
- 3 The supported output resolution is depend on the input board's type;
 - **HDMI/DVI/VGA/HDBST** output board support output resolution is as followed:
 "1024x768p60", "1280x720p60", "1280x1024p60", "1360x768p60",
 "1600x1200p60", "1680x1050p60", "1920x1080p30", "1920x1080p60",
 - **CVBS** output board support output resolution is as followed:
 "NTSC", "PAL"
 - **YUV** output board support output resolution is as followed:
 "1280x720p60", "1920x1080p60"
 - **SDI** output board support output resolution is as followed:
 "1280x720p60", "1920x1080p30", "1920x1080p60"
 - **UHD** output board support output resolution is as followed:
 "1280x720p50", "1280x720p60", "1920x1080p50", "1920x1080p60",
 "4K2Kp25", "4K2Kp30", "4K2Kp50", "4K2Kp60",
 "1024x768p60", "1280x768p60", "1280x1024p60", "1360x768p60",
 "1366x768p60", "1400x1050p60", "1600x1200p60", "1920x1200p60",
- 4 The SMS88-18G support output resolution as follows:
 3840x2160p60
 3840x2160p50
 3840x2160p30
 3840x2160p25
 1920x1200p60
 1920x1080p60
 1920x1080p50
 1600x1200p60
 1400x1050p60

1366x768p60
 1360x768p60
 1280x1024p60
 1280x768p60
 1280x720p50
 1280x720p60
 1024x768p60

2.4 Brightness Setting

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1 byte)	Command (10 bytes)	Command tail (1B)
GET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	BRIGHTNESS	↵ This is ASCII carriage return 0x0d

GET brightness of output board:

Send: GET OUT1 BRIGHTNESS↵

Receive: OUT1 BRIGHTNESS 50

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1 byte)	Command (10 bytes)	Command parameters (1B/2B/3B)	Command tail (1B)
SET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	BRIGHTNESS	x/xx/xxx x is the brightness value	↵ This is ASCII carriage return 0x0d

SET brightness of output board:

Send: SET OUT1 BRIGHTNESS 55↵

Receive: OUT1 BRIGHTNESS 55

2.5 Contrast Setting

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (8B)	Command tail (1B)
GET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	CONTRAST	↵ This is ASCII carriage return 0x0d

GET contrast of output board:

Send: GET OUT1 CONTRAST↵

Receive: OUT1 CONTRAST 50

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (8B)	Command parameters (1B/2B/3B)	Command tail (1B)
-------------------	----------------	----------------------	----------------	-----------------	----------------------------------	----------------------

SET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	CONTRAST	x/xx/xxx x is the contrast value	↵ This is ASCII carriage return 0x0d
-----	-------	--	-------	----------	-------------------------------------	---

SET contrast of input board:

Send: SET OUT1 CONTRAST 55↵

Receive: OUT1 CONTRAST 55

2.6 Saturation Setting

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (10B)	Command tail (1B)
GET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	SATURATION	↵ This is ASCII carriage return 0x0d

GET saturation of output board:

Send: GET OUT1 SATURATION↵

Receive: OUT1 SATURATION 50

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (10B)	Command parameters (1B/2B/3B)	Command tail (1B)
SET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	SATURATION	x/xx/xxx x is the saturation value	↵ This is ASCII carriage return 0x0d

SET saturation of output board:

Send: SET OUT1 SATURATION 55↵

Receive: OUT1 SATURATION 55

2.7 Sharpness Setting

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (10B)	Command tail (1B)
GET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	SHARPNESS	↵ This is ASCII carriage return 0x0d

GET sharpness of output board:

Send: GET OUT1 SHARPNESS↵

Receive: OUT1 SHARPNESS 50

Operation	Spacer	Target	Spacer	Command	Command	Command tail
-----------	--------	--------	--------	---------	---------	--------------

(3B)	(1B)	(4B/5B/6B)	(1B)	(10B)	parameters (1B/2B/3B)	(1B)
SET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	SHARPNESS	x/xx/xxx x is the sharpness value	↵ This is ASCII carriage return 0x0d

SET sharpness of output board:

Send: SET OUT1 SHARPNESS 55↵

Receive: OUT1 SHARPNESS 55

2.8 Picture Quality Reset

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (8B)	Command tail (1B)
SET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	PQ-RESET	↵ This is ASCII carriage return 0x0d

Reset the picture quality of output board:

Send: SET OUT1 PQ-RESET↵

Receive: OUT1 PQ-RESET

2.9 Software Version

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (7B)	Command tail (1B)
GET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	VERSION	↵ This is ASCII carriage return 0x0d

Get the software version of output board:

Send: GET OUT1 VERSION↵

Receive: OUT1 VERSION 2019/01/01-12:00:00

2.10 Set TV-WALL

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (6B)	Command parameters (6B)	Command tail (1B)
SET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	TVWALL	Line: Column: P: Q: Adjust(H4bit-H,L4bit-V) Input:	↵ This is ASCII carriage return 0x0d

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Picture for example: The entire TV wall consists of 16 screens, placed in 4 rows and 4 columns. Screens 6/7/10/11 make up a 2x2 splice.

The parameter of the splice which make up by Screens 6/7/10/11:

Line: How many rows of the Digital Information Display, picture for example, 2

Column: How many columns of the Digital Information Display left picture for example, 2

P: The row number of the current output connected: Screen 6: 1 Screen 7: 1 Screen 10: 2 Screen 11: 2

Q: The column number of the current output connected: Screen 6: 1 Screen 7: 2 Screen 10: 1 Screen 11: 2

The border of each screen is 20 pixels for example:

Adjust: High 4 bit: H_adjust; Low 4 bit: V_adjust

Input: Which input route to the current panel

A. SET TV-WALL mode of one output port

Picture Screen 6/7/10/11, and the source input is input 1 For example:

Send: SET OUT6 TVWALL 2 2 1 1 0 1 ↵

Receive: OUT6 TVWALL 2 2 1 1 0 1

Send: SET OUT7 TVWALL 2 2 1 2 0 1 ↵

Receive: OUT7 TVWALL 2 2 1 2 0 1

Send: SET OUT10 TVWALL 2 2 2 1 0 1 ↵

Receive: OUT10 TVWALL 2 2 2 1 0 1

Send: SET OUT11 TVWALL 2 2 2 2 0 1 ↵

Receive: OUT11 TVWALL 2 2 2 2 0 1

Sending these four commands will create a 2x2 splice

B. How to Exit TV wall mode:

e.g Exit TV-WALL combination of output port6,7,10,11

Send: SET OUT6 TVWALL 1 1 1 1 0 1 ↵

Receive: OUT6 TVWALL 1 1 1 1 0 1

Send: SET OUT7 TVWALL 1 1 1 1 0 1 ↵

Receive: OUT7 TVWALL 1 1 1 1 0 1

Send: SET OUT10 TVWALL 1 1 1 1 0 1 ↵

Receive: OUT10 TVWALL 1 1 1 1 0 1

Send: SET OUT11 TVWALL 1 1 1 1 0 1↵

Receive: OUT11 TVWALL 1 1 1 1 0 1

2.11 Read EDID

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (4B)	Command parameters (N bytes)	Command tail (1B)
GET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	EDID	Send: (5B/6B) PART1 PART2 PART16 Receive: (53/54 bytes) PART1 d1 d2 ... d16 PART2 d1 d2 ... d16 PART16 d1 d2 ... d16	↵ This is ASCII carriage return 0x0d

A. GET (Read) EDID data from output port

For example, GET (Read) EDID data from output port 1 (Read 16 times)

Send: GET OUT1 EDID

Receive: OUT1 EDID PART1 00 FF FF FF FF FF FF 00 63 18 22 00 66 00 00 00

Receive: OUT1 EDID PART2 05 1C 01 03 80 59 32 78 0A EE 91 A3 54 4C 99 26

Receive: OUT1 EDID PART3 0F 50 54 01 08 00 81 C0 81 C0 81 00 81 80 95 00

Receive: OUT1 EDID PART4 A9 C0 B3 00 01 01 08 E8 00 30 F2 70 5A 80 B0 58

Receive: OUT1 EDID PART5 8A 00 C4 8E 21 00 00 1E 02 3A 80 18 71 38 2D 40

Receive: OUT1 EDID PART6 58 2C 45 00 50 1D 74 00 00 1E 00 00 00 FD 00 17

Receive: OUT1 EDID PART7 3C 0F 88 3C 00 0A 20 20 20 20 20 20 00 00 00 FC

Receive: OUT1 EDID PART8 00 48 44 4D 49 0A 20 20 20 20 20 20 20 01 16

Receive: OUT1 EDID PART9 02 03 40 F1 55 61 10 1F 04 13 05 14 20 21 22 5D

Receive: OUT1 EDID PART10 5E 5F 60 65 66 07 12 03 16 01 23 09 07 07 83 01

Receive: OUT1 EDID PART11 00 00 6E 03 0C 00 30 00 B8 3C 21 10 80 01 02 03

Receive: OUT1 EDID PART12 04 67 D8 5D C4 01 78 80 03 E2 00 4F E3 0F 01 E0

Receive: OUT1 EDID PART13 01 1D 80 D0 72 1C 16 20 10 2C 25 80 50 1D 74 00

Receive: OUT1 EDID PART14 00 9E 66 21 56 AA 51 00 1E 30 46 8F 33 00 50 1D

Receive: OUT1 EDID PART15 74 00 00 1E 00 00 00 00 00 00 00 00 00 00 00

Receive: OUT1 EDID PART16 00 00 00 00 00 00 00 00 00 00 00 00 00 00 A1

NOTE:

1. EDID totally have 256 bytes, so we need to read 16 times and 16 bytes will be read per time.

3.System command

3.1 Device size

Operation (3B)	Spacer (1B)	Target (3B)	Spacer (1B)	Command (4B)	Command tail (1B)
GET	Space	SYS	Space	SIZE	↵ This is ASCII carriage return 0x0d

GET the device size:

For example, GET the device size (8x8)

Send: GET SYS SIZE↵

Receive: SYS SIZE 8 8 24

3.2 Device IP

Operation (3B)	Spacer (1B)	Target (3B)	Spacer (1B)	Command (2B)	Command tail (1B)
GET	Space	SYS	Space	IP	↵ This is ASCII carriage return 0x0d

GET the device size:

For example, GET the device IP

Send: GET SYS IP↵

Receive: SYS IP DHCP,192.168.0.119,255.255.255.0,192.168.0.1

Send: GET SYS IP↵

Receive: SYS IP STATIC,192.168.0.222,255.255.255.0,192.168.0.24

Operation (3B)	Spacer (1B)	Target (3B)	Spacer (1B)	Command (2B)	Command parameters (NB)	Command tail (1B)
SET	Space	SYS	Space	IP	mode,address,mask,gateway	↵ This is ASCII carriage return 0x0d

SET the device IP:

For example, Set the device IP to STATIC 192.168.0.222

Send: SET SYS IP STATIC,192.168.0.222,255.255.255.0,192.168.0.1↵

Receive: SYS IP STATIC,192.168.0.222,255.255.255.0,192.168.0.1

For example, Set the device IP to DHCP (auto obtain)

Send: SET SYS IP DHCP↵

Receive: SYS IP DHCP

3.3 System Reset

Operation (3B)	Spacer (1B)	Target (3B)	Spacer (1B)	Command (5B)	Command parameters (3B)	Command tail (1B)
SET	Space	SYS	Space	RESET	ALL	↵ This is ASCII carriage return 0x0d

SET (Reset) the device :

For example:

Send: SET SYS RESET ALL↵

Receive: SYS RESET ALL

3.4 Panel Lock

4. Operation 5. (3B)	Spacer (1B)	Target (3B)	Spacer (1B)	Command (10B)	Command tail (1B)
GET	Space	SYS	Space	PANEL-LOCK	↵ This is ASCII carriage return 0x0d

GET the panellock

For example:

Send: GET SYS PANEL-LOCK ↵

Receive: SYS PANEL-LOCK ON

Send: GET SYS PANEL-LOCK ↵

Receive: SYS PANEL-LOCK OFF

Operation (3B)	Spacer (1B)	Target (3B)	Spacer (1B)	Command (10B)	Command parameters (2B/3B)	Command tail (1B)
SET	Space	SYS	Space	PANEL-LOCK	ON or OFF	↵ This is ASCII carriage return 0x0d

SET the panellock

For example:

Send: SET SYS PANEL-LOCK ON↵

Receive: SYS PANEL-LOCK ON

Send: SET SYS PANEL-LOCK OFF↵

Receive: SYS PANEL-LOCK OFF

3.5 System Version

Operation (3B)	Spacer (1B)	Target (3B)	Spacer (1B)	Command (10B)	Command tail (1B)
GET	Space	SYS	Space	VERSION	↵ This is ASCII carriage return 0x0d

Get the system version

For example:

Send: GET SYS VERSION ↵

Receive: SYS VERSION 2020/9/2-11:11:54

3.6 TVWALL Mode

Operation (3B)	Spacer (1B)	Target (3B)	Spacer (1B)	Command (10B)	Command parameters (1B/2B/3B)	Command tail (1B)
GET	Space	SYS	Space	TVWALL-MODE	x/xx/xxx x is the mode value	↵ This is ASCII carriage return 0x0d

GET (Recall) the route mode saved before:

For example, GET (Recall) the route mode 1

Send: GET SYS TVWALL-MODE 1↵

Receive: SYS TVWALL-MODE 1

Operation (3B)	Spacer (1B)	Target (3B)	Spacer (1B)	Command (10B)	Command parameters (1B/2B/3B)	Command tail (1B)
SET	Space	SYS	Space	TVWALL-MODE	x/xx/xxx x is the mode value	↵ This is ASCII carriage return 0x0d

SET (Save) current route to a mode:

For example, SET (Save) current route to mode 1

Send: SET SYS TVWALL-MODE 1↵

Receive: SYS TVWALL-MODE 1