

MMS-xxx RS232 ASCII Protocol

Version	V2.3
Date	2022-11-07

Content

MMS-xxx 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	6
1.8 Input Board Audio Select	7
2 Output Board Command	8
2.1 Input Signal format	8
2.2 Output Type.....	9
2.3 Output Signal format	11
2.4 Brightness Setting.....	17
2.5 Contrast Setting	17
2.6 Saturation Setting.....	18
2.7 Sharpness Setting	18
2.8 Picture Quality Reset	19
2.9 Software Version	19
2.10 Set TV-WALL	19
2.11 Read EDID	21
3. System command	22
1 Device size	22
2 Device IP	22
3 System Reset	22
4 Panel Lock.....	23
5 System Version	23
6 TVWALL Mode	24
3.7 Matrix Route Info	24
4.add command.....	24
4.1 Host model	24
4.2 Host SN	25
4.3 Main Chip temperature	25
4.4 tcp subnet mask.....	25
4.5 tcp geteway	26
4.6 hardware input port connect status.....	26
4.7 hardware output port connect status	26
4.8 produce time	26

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

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↵

B. Set multichannel video route (!!!At most, only 8 outputs can be switched at the same time!!!):

For example: Input port 1 switch to output port 1,2,3,4,5,6,7,8

Send: SET IN1 VIDEO OUT1,2,3,4,5,6,7,8↵

Receive: IN1 VIDEO OUT1,2,3,4,5,6,7,8↵

For example: Input port 1 switch to output port 11, 6,30,8

Send: SET IN1 VIDEO OUT11, 6,30,8↵

Receive: IN1 VIDEO OUT11, 6,30,8↵

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

Get input board input type:

Send: GET IN1 IN-TYPE↵

Receive: IN1 IN-TYPE HDMI BOARD-TYPE HDMI↵

Send: GET IN2 IN-TYPE↵

Receive: IN1 IN-TYPE DIRECTHDMI BOARD-TYPE 4KHDMI↵

Send: GET IN3 IN-TYPE↵

Receive: IN3 IN-TYPE DVI BOARD-TYPE DVIU↵

Send: GET IN5 IN-TYPE↵

Receive: IN5 IN-TYPE HDBST BOARD-TYPE 4KHDBST↵

Send: GET IN7 IN-TYPE↵

Receive: IN7 IN-TYPE SDI BOARD-TYPE SDI↵

note:

1) all signal type as follow:

FIBER
SDI
HDBST
DVI
CVBS
YPBPR
VGA
HDMI
DIRECTHDMI
UHDHDMI
UHDDVI
UHDHDMI14
UHDHDMI22

2) all input board type as follow:

HDMI
DVIU

4KHDBST
SDI
4KHDMI
4KFIBER
2KHDBST
2KFIBER
DIRECT-IN
4KHDMI-6265

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

Get input board input signal format

```
Send: GET IN1 IN-SIGNAL↵
Receive: IN1 IN-SIGNAL @3840x2160p30↵
Send: GET IN2 IN-SIGNAL↵
Receive: IN2 IN-SIGNAL @3840x2160p30↵
Send: GET IN3 IN-SIGNAL↵
Receive: IN3 IN-SIGNAL @1920x1080p60↵
Send: GET IN5 IN-SIGNAL↵
Receive: IN5 IN-SIGNAL @1920x1080p60↵
Send: GET IN7 IN-SIGNAL↵
Receive: IN7 IN-SIGNAL @NO-SIGNAL↵
```

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

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

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:

1. The output type of input board only support DVI and HDMI

2. only support GET, not support SET

Software Version

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

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↵
```

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.

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)
SET	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

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 @1920x1080p60↵

Send: GET OUT3 IN-SIGNAL↵

Receive: OUT3 IN-SIGNAL @NO-SIGNAL↵

NOTE:

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

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 board output signal type:

Send: GET OUT1 OUT-TYPE↵

Receive: OUT1 OUT-TYPE HDMI BOARD-TYPE HDMI↵

Send: GET OUT3 OUT-TYPE↵

Receive: OUT5 OUT-TYPE HDBST BOARD-TYPE 2KHDBST↵

Send: GET OUT5 OUT-TYPE↵

Receive: OUT7 OUT-TYPE SDI BOARD-TYPE SDI↵

Note:

1)all signal type as follow:

FIBER
SDI
HDBST
DVI
CVBS
YPBPR
VGA
HDMI
DIRECTHDMI
UHDHDMI
UHDDVI
UHDHDMI14
UHDHDMI22

2)all output board type as follow:

HDMI
DVIU
4KHDBST
SDI
HDMI1080i
DVIU1080i
4KHDMI
4KFIBER
M88-OUT
2KHDBST
2KFIBER
4KHDMI-6265
YPBPR-1080i
MST9804-6265
4KHDMI-6265-OUT

3)In the message `OUT5 OUT-TYPE HDBST BOARD-TYPE 2KHDBST`, the 2KHDBST representative after the `BOARD-TYPE` is the 2K HDBST output board

Operation (3B)	Spacer (1B)	Target (4B/5B/6B)	Spacer (1B)	Command (8B)	Command parameters (NB)	Command tail (1B)
SET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	OUT-TYPE	CVBS/VGA/YUV/HDMI/DVI	↵ This is ASCII carriage return 0x0d

Set output board output signal type:

Send: SET OUT1 OUT-TYPE UHDHDMI↵
Receive: OUT1 OUT-TYPE UHDHDMI BOARD-TYPE 4KHDMI↵
Send: SET OUT3 OUT-TYPE DVI↵
Receive: OUT01 OUT-TYPE DVI BOARD-TYPE HDMI↵
Send: SET OUT3 OUT-TYPE CVBS↵
Receive: OUT3 OUT-TYPE CVBS BOARD-TYPE DVIU↵
Send: SET OUT3 OUT-TYPE YPBPR↵
Receive: OUT3 OUT-TYPE YPBPR BOARD-TYPE DVIU↵
Send: SET OUT3 OUT-TYPE VGA↵
Receive: OUT3 OUT-TYPE VGA BOARD-TYPE DVIU↵
Send: SET OUT3 OUT-TYPE HDMI↵
Receive: OUT3 OUT-TYPE HDMI BOARD-TYPE HDMI↵

Note:

1) all signal type as follow:

FIBER
SDI
HDBST
DVI
CVBS
YPBPR
VGA
HDMI
DIRECTHDMI
UHDHDMI
UHDDVI

UHDHDMI14
 UHDHDMI22
 2) all output board type as follow:
 HDMI
 DVIU
 4KHDBST
 SDI
 HDMI1080i
 DVIU1080i
 4KHDMI
 4KFIBER
 M88-OUT
 2KHDBST
 2KFIBER
 4KHDMI-6265
 YPBPR-1080i
 MST9804-6265
 4KHDMI-6265-OUT

3) different boards support different type parameters ,Refer to 4) for details

4)Output type cannot be set when output port X is FIBER/SDI/HSBST/DIRECTHDMI type, other settings need to follow the following operation

4.1)before setting the output type of output port X each time, we need to obtain the output type of output port X and get the board type of output port X

Send: GET OUTx OUT-TYPE↵

4.2)The following is a description of the output types that can be set for each board

- ①The board type is HDMI/HDMI1080i/M88-OUT. Only HDMI / DVI can be set as the output type
- ②The board type is DVIU/DVIU1080i, and only the output type can be set as HDMI/DVI/VGA/CVBS/YPBPR
- ③The board type is 4KHDBST/4KFIBER/SDI/2KHDBST/2KFIBER/YPBPR-1080i and the output type cannot be modify
- ④The board type is 4KHDMI/4KHDMI-6265/MST9804-6265/4KHDMI-6265-OUT, only the output type can be set as UHDHDMI/UHDDVI/UHDHDMI14/UHDHDMI22

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 return 0x0d

GET output signal format of output board:

Send: GET OUT1 OUT-SIGNAL↵
 Receive: OUT1 OUT-SIGNAL @3840x2160p30↵
 Send: GET OUT3 OUT-SIGNAL↵
 Receive: OUT3 OUT-SIGNAL @1920x1080p60↵
 Send: GET OUT5 OUT-SIGNAL↵
 Receive: OUT5 OUT-SIGNAL @1920x1200p60↵
 Send: GET OUT7 OUT-SIGNAL↵
 Receive: OUT7 OUT-SIGNAL @1280x0720p60↵

Operation type (3B)	Spacer (1B)	Target (4B/5B/6B)	Space r (1B)	Command (10B)	Command parameters (NB)	Command tail (1B)
SET	Space	OUTx/OUTxx/OUTxxx x is the output port number	Space	OUT-SIGNAL	TYPE@Resolution	↵ This is ASCII carriage return 0x0d

SET output signal format of output board:

Send: SET OUT1 OUT-SIGNAL 1024x0768p60↵
 Receive: OUT1 OUT-SIGNAL @1024x0768p60↵
 Send: SET OUT3 OUT-SIGNAL 1024x0768p60↵
 Receive: OUT01 OUT-SIGNAL @1024x0768p60↵
 Send: SET OUT3 OUT-SIGNAL 1280x0720p60↵
 Receive: OUT3 OUT-SIGNAL @1280x0720p60↵
 Send: SET OUT3 OUT-SIGNAL 1280x1024p60↵

Receive: OUT3 OUT-SIGNAL @1280x1024p60↵
Send: SET OUT3 OUT-SIGNAL 1360x0768p60↵
Receive: OUT3 OUT-SIGNAL @1360x0768p60↵

Note:

1) all signal type as follow:

FIBER
SDI
HDBST
DVI
CVBS
YPBPR
VGA
HDMI
DIRECTHDMI
UHDHDMI
UHDDVI
UHDHDMI14
UHDHDMI22

2) all output board type as follow:

HDMI
DVIU
4KHDBST
SDI
HDMI1080i
DVIU1080i
4KHDMI
4KFIBER
M88-OUT
2KHDBST
2KFIBER
4KHDMI-6265
YPBPR-1080i
MST9804-6265
4KHDMI-6265-OUT

3) different boards support different type parameters. See 4) for details

4) Output type cannot be set when output port X is FIBER/SDI/HSBST/DIRECTHDMI type, other settings need to follow the following operation

4.1) Each time before setting the output resolution of output x, we need to obtain the output type of output X and get the board type of output X

Send: GET OUTx OUT-TYPE↵

4.2) The following is a description of the output types that can be set for each board

4.2.1)The board type is HDMI /M88-OUT, and the resolution can be set as follows:

"1024x0768p60",
"1280x0720p60",
"1280x1024p60",
"1360x0768p60",
"1600x1200p60",
"1680x1050p60",
"1920x1080p30",
"1920x1080p60",
"1280x0720p50",
"1920x1080p50",
"1920x1200p60",

4.2.2)The board type is DVIU. If the output type is CVBS, the resolution can be set as follows:

"NTSC",
"PAL",

4.2.3)The board type is DVIU. If the output type is YPbPr, the resolution can be set as follows:

"1280x0720p60",
"1920x1080p60",

4.2.4)The board type is DVIU. If the output type is VGA, the resolution can be set as follows:

"1024x0768p60",
"1280x0720p60",
"1280x1024p60",
"1360x0768p60",
"1600x1200p60",
"1680x1050p60",
"1920x1080p60",
"1920x1200p60",

4.2.5)The board type is DVIU. If the output type is HDMI / DVI, the resolution can be set as follows:

"1024x0768p60",
"1280x0720p60",

"1280x1024p60",
"1360x0768p60",
"1600x1200p60",
"1680x1050p60",
"1920x1080p30",
"1920x1080p60",
"1280x0720p50",
"1920x1080p50",
"1920x1200p60",

4.2.6)The board type is **HDMI1080i**, and the resolution can be set as follows:

"1024x0768p60",
"1280x0720p60",
"1600x1200p60",
"1680x1050p60",
"1920x1080p30",
"1920x1080p60",
"1280x0720p50",
"1920x1080p50",
"1920x1200p60",
"1920x1080i50",
"1920x1080i60",

4.2.7)The board type is **DVIU1080i**. If the output type is CVBS, the resolution can be set as follows:

"NTSC",
"PAL",

4.2.8) The board type is **DVIU1080i**. If the output type is YPBPR, the resolution can be set as follows:

"1280x0720p60",
"1920x1080p60",

4.2.9) The board type is **DVIU1080i**. If the output type is VGA, the resolution can be set as follows:

"1024x0768p60",
"1280x0720p60",
"1600x1200p60",
"1680x1050p60",
"1920x1080p60",
"1920x1200p60",

4.2.10) The board type is **DVIU1080i**. If the output type is HDMI/DVI, the resolution can be set as follows:

"1024x0768p60",
"1280x0720p60",
"1600x1200p60",
"1680x1050p60",
"1920x1080p30",
"1920x1080p60",
"1280x0720p50",
"1920x1080p50",
"1920x1200p60",
"1920x1080i50",
"1920x1080i60",

4.2.11)The board type is **SDI**, and the resolution can be set as follows:

"1920x1080p60",
"1920x1080p50",
"1920x1080p30",
"1920x1080p25",
"1920x1080p24",
"1920x1080i60",
"1920x1080i50",
"1280x0720p60",
"1280x0720p50",
"1280x0720p30",
"1280x0720p25",
"0720x0480i60",
"0720x0576i50",
"AUTO", (The output resolution is consistent with the input resolution)

4.2.12)The board type is **4KHDMI/4KHDBST/4KFIBER**, and the resolution can be set as follows:

"1280x0720p50",
"1280x0720p60",
"1920x1080p50",
"1920x1080p60",
"3840x2160p25",
"3840x2160p30",
"3840x2160p50",
"3840x2160p60",
"1024x0768p60",
"1280x0768p60",

"1280x1024p60",
"1360x0768p60",
"1366x0768p60",
"1400x1050p60",
"1600x1200p60",
"1920x1200p60",

4.2.13) The board type is **2KHDBST/2KFIBER**, and the resolution can be set as follows:

"1280x0720p50",
"1280x0720p60",
"1920x1080p50",
"1920x1080p60",
"1024x0768p60",
"1280x0768p60",
"1280x1024p60",
"1360x0768p60",
"1366x0768p60",
"1400x1050p60",
"1600x1200p60",
"1920x1200p60",

4.2.14) The board type is **YPBPR-1080i**, and the resolution can be set as follows:

"1280x0720p50",
"1280x0720p60",
"1920x1080p50",
"1920x1080p60",
"1920x1080i50",
"1920x1080i60",

4.2.15) The board type is **4KHDMI-6265**, and the resolution can be set as follows:

"1280x0720p50",
"1280x0720p60",
"1920x1080p50",
"1920x1080p60",
"3840x2160p25",
"3840x2160p30",
"3840x2160p50",
"3840x2160p60",
"1024x0768p60",
"1280x0768p60",
"1280x1024p60",
"1360x0768p60",
"1366x0768p60",
"1400x1050p60",
"1600x1200p60",
"1920x1200p60",
"4096x2160p60",
"4096x2160p50",

4.2.16) The board type is **MST9804-6265**, and the resolution can be set as follows:

"1280x0720p50",
"1280x0720p60",
"1920x1080p50",
"1920x1080p60",
"3840x2160p25",
"3840x2160p30",
"3840x2160p50",
"3840x2160p60",
"1024x0768p60",
"1280x0800p60",
"1280x1024p60",
"1360x0768p60",
"1920x1200p60",
"4096x2160p60",
"4096x2160p50",

4.2.17) The board type is **4KHDMI-6265-OUT**, and the resolution can be set as follows:

"1280x0720p50",
"1280x0720p60",
"1920x1080p50",
"1920x1080p60",
"3840x2160p25",
"3840x2160p30",
"3840x2160p50",
"3840x2160p60",
"1024x0768p60",
"1280x0768p60",
"1280x1024p60",

"1360x0768p60",
 "1366x0768p60",
 "1400x1050p60",
 "1600x1200p60",
 "1920x1200p60",
 "3440x1440p60",
 "2560x1600p60",
 "2560x1440p60",

Above all resolutions, horizontal and vertical effective pixels keep 4 characters on them!!!

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↵

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↵

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↵

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 (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	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↵

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 ↵

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↵

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↵

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

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↵

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↵

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↵

Panel Lock

Operation (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↵

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↵

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↵

Matrix Route Info

Operation (3B)	Spacer (1B)	Target (3B)	Spacer (1B)	Command (12B)	Command tail (1B)
GET	Space	SYS	Space	ROUTE-MATRIX	↵ This is ASCII carriage return 0x0d

Send: GET SYS ROUTE-MATRIX↵

Receive: SYS ROUTE-MATRIX 001 002 003 004 006 005 007 008↵

analyze:

The valid output port 1~8 corresponds to the input port of 001 002 003 004 006 005 007 008

4.add command

4.1 Host model

Operation (3B)	space (1B)	Target (7B)	space (1B)	Command (4B)	Command tail (1B)
GET	空格	DEVICES	space	NAME	↵ This is ASCII carriage return 0x0d

Send: GET DEVICES NAME↵

Receive: DEVICES NAME HDP-MMS-xxx↵

Operation (3B)	space (1B)	Target (7B)	space (1B)	Command (4B)	Command tail (1B)
SET	space	DEVICES	space	NAME	↵ This is ASCII carriage return 0x0d

Send: SET DEVICES NAME HDP-MX91616VM↵

Receive: DEVICES NAME HDP-MX91616VM↵

4.2 Host SN

Operation (3B)	space (1B)	Target (7B)	space (1B)	Command (2B)	Command tail (1B)
GET	space	DEVICES	space	SN	↵ This is ASCII carriage return 0x0d

Send: GET DEVICES SN↵

Receive: DEVICES SN G22DIM0001↵

Operation (3B)	space (1B)	Target (7B)	space (1B)	Command (2B)	Command tail (1B)
SET	space	DEVICES	space	SN	↵ This is ASCII carriage return 0x0d

Send: SET DEVICES SN G22DJM0011↵

Receive: DEVICES SN G22DJM0011↵

4.3 Main Chip temperature

Operation (3B)	space (1B)	Target (3B)	space (1B)	Command (11B)	Command tail (1B)
GET	space	MCU	space	temperature	↵ This is ASCII carriage return 0x0d

Send: GET MCU temperature↵

Receive: MCU temperature 41.25↵

4.4 tcp subnet mask

Operation (3B)	space (1B)	Target (6B)	space (1B)	Command (4B)	Command tail (1B)
GET	space	SUBNET	space	MASK	↵ This is ASCII carriage return 0x0d

Send: GET SUBNET MASK↵

Receive:SUBNET MASK 255.255.255.0↵

4.5 tcp gateway

Operation (3B)	space (1B)	Target (6B)	space (1B)	Command (2B)	Command tail (1B)
GET	space	GATWAY	space	IP	↵

					This is ASCII carriage return 0x0d
--	--	--	--	--	------------------------------------

Send: GET GATWAY IP↵

Receive: GATWAY IP 192.168.0.1↵

4.6 hardware input port connect status

Operation (3B)	space (1B)	Target (2B)	space (1B)	Port (1B/2B/3B)	space (1B)	Command (5B)	Command tail (1B)
GET	space	IN	space	x/xx/xxx x is the input port number	space	STATE(0-disconnect,1-connect)	↵ This is ASCII carriage return 0x0d

Send: GET IN 5 STATE↵

Receive: IN 5 0↵

Send: GET IN 5 STATE↵

Receive: IN 5 1↵

4.7 hardware output port connect status

Operation (3B)	space (1B)	Target (3B)	space (1B)	Port (1B/2B/3B)	space (1B)	Command (5B)	Command tail (1B)
GET	space	OUT	space	x/xx/xxx x is the output port number	space	STATE(0-disconnect,1-connect)	↵ This is ASCII carriage return 0x0d

Send: GET OUT 5 STATE↵

Receive: OUT 5 0↵

Send: GET OUT 5 STATE↵

Receive: OUT 5 1↵

4.8 produce time

Operation (3B)	space (1B)	Target (7B)	space (1B)	Command (4B)	Command tail (1B)
GET	space	PRODUCE	space	TIME	↵ This is ASCII carriage return 0x0d

Send: GET PRODUCE TIME↵

Receive: PRODUCE TIME 202209↵