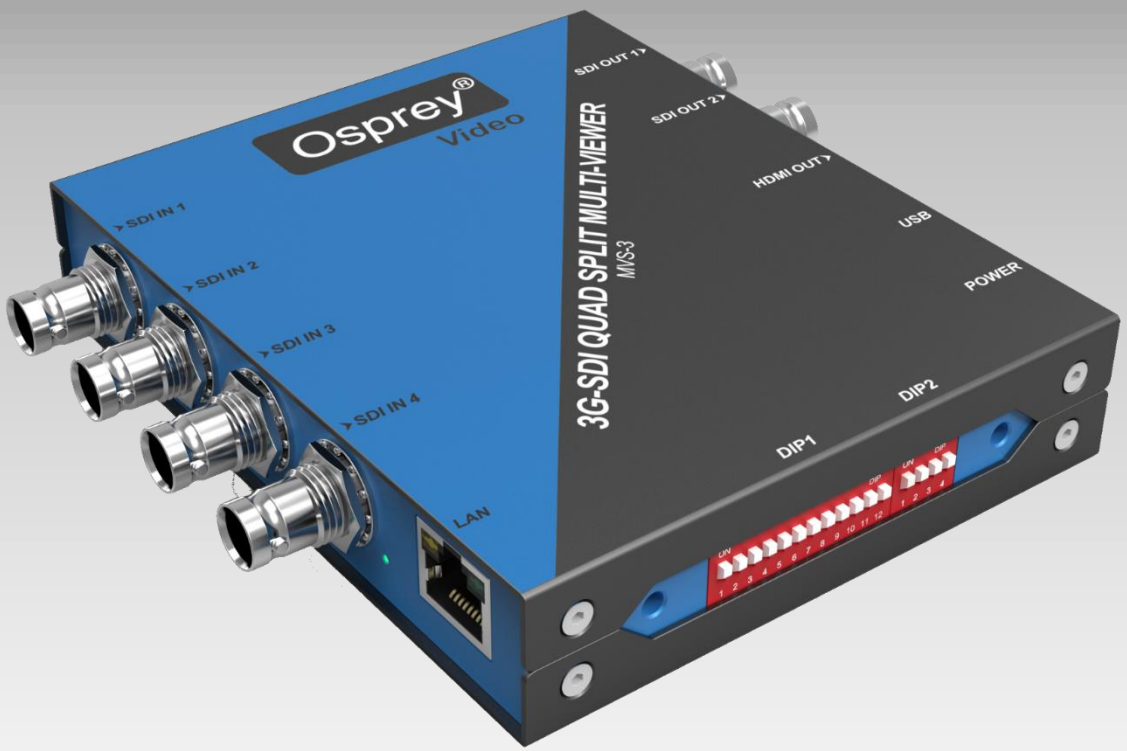
Osprey® Video

API Guide MVS-3



Configuration and Control API Guide for LAN

Communication:
Interface: LAN
Communication Protocol: UDP Broadcast
Destination Port: 7000

www.ospreyvideo.com

Broadcast from PC to MVS-3:

Data Packet	Value	Byte	Description
Packet Header	0xA5 0x6C	2	The beginning of data packet.
Data Length	0x0000~0x0420	2	The length of the entire data packet from packet header to end (including the packet header and end). The lower byte stays ahead.
Device Type	0x00~0xFF	1	Definition of device type, OXFF means broadcast.
Device ID	0x00~0xFF	1	A distinguishing of the device when there are several devices in a same LAN at same time. OXFF means broadcast.
Interface Type	0x00~0xFF	1	0x00:UART（serial port) 0x01: LAN
Reserve	0x00	9	For reserve. This device is not reserved.
Command	0x00~0xFF	1	Command for each function.
Packet Data	.....	Indefinite	<= 1024
Checksum	0x0000~0xFFFF	2	The algebraic sum of all bytes from packet header to checksum, in 2 bytes, other parts omitted ((including the packet header and checksum)). The lower byte stays ahead.
Packet End	0xAE	1	The end of the packet.

Response from MVS-3:

Data Packet	Value	Byte	Description
Packet Header	0xA5 0x6C	2	The beginning of data package.
Data Length	0x0000~0xFFFF	2	The length of the entire data packet from packet header to end (including the packet header and end). The lower byte stays ahead.
Device Type	0x00~0xFF	1	Definition of device type, OXFF means broadcast
Device ID	0x00~0xFF	1	A distinguishing of the device when there are several devices in a same LAN at same time. OXFF means broadcast.
Interface Type	0x00~0xFF	1	0x00: UART（serial port); 0x01: LAN
Reserve	0x00	9	Reserve. This device is not reserved.
Command	0x00~0xFF	1	Command for each function.
Response Status	0x00 ~ 0xFF	1	0x00： Succeed; 0x01: Error; Other data undefined.
Response Content		Indefinite	Reserve. The length of response content is variable when backward reading command, and it is consistent with the format of "packet data".
Checksum	0x0000~0xFFFF	2	The algebraic sum of all bytes from packet header to checksum, in 2 bytes, other parts omitted. The lower byte stays ahead.
Packet End	0xAE	1	The end of the packet.

Note: Broadcast CMD+ data; Response CMD+ status+ data

Commands:

Device Type: 0xa3

Function	Command（hex）	Description
Scanning	0xff	Broadcast to scan the multiviewer from the LAN.
Reading All the Data	0x90	After device scanned, reading all status data of the device.
Rename the Device	0x91	Change the name of device.
Output Resolution	0x92	Change the device output resolution.
Border Enable	0xa4	Turn on/off the border of windows.
Border color	0x93	Setting border color of windows.
Output Layout	0x94	Change the output layouts.
UMD Overlay Enable	0x95	Turn on/off the UMD overlay. 1: ON, 0: OFF
UMD Position	0x96	Change the location of the UMD. 0:left 1:center 2:right
UMD Character Color	0x97	Change the color of UMD text (refer to color list)
UMD Background Color	0x98	UMD background color (refer to color list)
UMD Text	0x99	Input the content of UMD text. 16 characters max length.
Audio Meter Enable	0x9a	Turn on/off the audio meter. 1: ON, 0: OFF
Audio Meter Position	0x9b	Change the position of audio meter. 0: left, 2: right
Audio Source	0x9e	Change the source of audio input (refer to the list).
OSD Enable	0x9f	Turn on/off the OSD. 1: ON, 0: OFF
OSD Text Color	0xa0	Change the OSD text color (refer to color list).
OSD Background Color	0xa1	Change the OSD background color (refer to color list).
OSD Position	0xa2	Change the positions of OSD. 0: left, 1: center, 2: right
Custom	0xa3	Save the current settings to custom 1 or custom 2. 1: custom 1, 2: custom 2
Size of Character	0xa5	Setting the size of overlay character. 0: small ,1: middle,2: large
Reset	0xa6	Reset all settings to factory settings, including IP address.
Manual setting IP	0xa7/0x05	12 bytes. including IP address, Subnet mask, gateway. “0xa7” or “0x05” are both works in same result.

List of Commands:

```
MV0430_READ_ALL_PAGE_DATA = 0x90,          /*Reading data*/ /*No parameters*/
MV0430_SEND_STATUS_CUSTOM_NAME,            /*Device name */ /*X Parameter (without line number)*/
MV0430_SEND_OUTPUT_FORMAT,                 /*Output resolution*//*1 parameter*/
MV0430_SEND_BORDER_COLOR,                  /* Border color*/ /*1 parameter*/
MV0430_SEND_OUTPUT_LAYOUT,                 /*Output layout*/ /*1 parameter*/
MV0430_SEND_OVERLAY_UMD_ENABLE,            /* UMD display enable*/ /*2 parameter*/
MV0430_SEND_OVERLAY_UMD_POS,               /*UMD position*/ /*2 parameter*/
MV0430_SEND_OVERLAY_TEXT_COLOR,            /*UMD text color*/ /*2 parameter*/
MV0430_SEND_OVERLAY_BACK_COLOR,            /*UMD background color*/ /*2 parameter*/
MV0430_SEND_OVERLAY_UMD_MSG,               /*UMD content*/ /*X parameter*/
MV0430_SEND_AUDIO_ENABLE,                  /*Audio meter enable*/ /*2 parameter*/
MV0430_SEND_AUDIO_POS,                     /*Audio meter position*/ /*2 parameter*/
MV0430_SEND_INPUT_ENABLE,                  /*OSD enable*/ /*2 parameter*/
MV0430_SEND_INPUT_TEXT_COLOR,              /*OSD text color*/ /*2 parameter*/
MV0430_SEND_INPUT_BACK_COLOR,              /*OSD background color*/ /*2 parameter*/
MV0430_SEND_INPUT_POS,                     /*OSD position*/ /*2 parameter*/
MV0430_SEND_SETTING_SET_CUSOTM,            /*Setting custom*/  /*1 parameter*/
MV0430_SEND_BORDER_ENABLE,                 /*Border enable*/ /*1 parameter*/
MV0430_SEND_SETTING_SET_UMD_FONT,          /*Setting character size */ /*1 parameter*/
MV0430_SEND_ALL_PAGE_DATA,                 /*reset（long package）*/ /*X  parameter*/
MV0430_SEND_SETTING_SET_DEV_IP,            /*Setting IP (reserved)）*/ /*12  parameter*/
```

Note: All commands with 2 parameter, and data part is “window ID + value”, serial ID is 0 1 2 3 in turn

Parameter List

1. Response Format

```
typedef struct
{
unsigned char  mv0430_ver_Fgpa;                 //fpga version  1byte
unsigned char  mv0430_ver_Mcu;                  //mcu version  1byte
unsigned char  mv0430_output_format;            //output resolution
unsigned char  mv0430_border_color;             //(xxx) border color
unsigned char  mv0430_border_enable;            //border enable
unsigned char  mv0430_output_layout;            //output layout
unsigned char  mv0430_umd_font_size;            //UMD character size
unsigned char  res_total[5];                    //reserve
char m_mv0430_custom_name[17];                  //Device name 16 characters+‘\0’
UMD_TOTAL_DATA   m_strcut_umd_data[4];          //page umd
AUDIO_TOTAL_DATA   m_strcut_audio_data[4];      //page audio
OSD_TOTAL_DATA   m_strcut_osd_data[4];          //page osd
}ALL_MV0430_CHILD_DLG_DATA;
```

```
Typedef struct
{
unsigned char  umd_enable;                  //umd enable
unsigned char  umd_pos;                     // UMD position
unsigned char  umd_text_color;              //UMD text color
unsigned char  umd_background_color;        //UMD background color
unsigned char  res_umd[3];                  //UMD reserved
unsigned char  umd_len;                     //UNM length
char  umd_str[32];                          //UMD  text
}UMD_TOTAL_DATA;

Typedef struct
{
unsigned char  osd_enable;                  //OSD enable
unsigned char  osd_pos;                     //OSD position
unsigned char  osd_text_color;              //OSD text color
unsigned char  osd_background_color;        //OSD background color
unsigned char  res_osd[2];                  //OSD reserved
}OSD_TOTAL_DATA;
```

2. Output Resolutions

Output Resolution	Broadcast Value
1080p60	0
1080p50	1
1080p30	2
1080p25	3
1080p24	4
1080i60	5
1080i50	6
720p60	7
720p50	8
720p30	9
720p25	10

3. Layouts

Layout	Broadcast Value
	0
	1
	2
	3
	4
	5
	6
	7
	8
	9
	10
	11
	12
	13
	14
	15

4. Audio Channels

Channel	Broadcast Value
CH 1-2	0
CH 3-4	1
CH5-6	2
CH7-8	3
CH9-10	4
CH11-12	5
CH13-14	6
CH15-16	7

5. Colors

Color	Broadcast Value
Black	0
Blue	1
Red	2
Megenta	3
Green	4
Cyan	5
Yellow	6
White	7
Gray	8
VioletRed	9
LightBlue	10
LightGreen	11
LightCyan	12
LightYellow	13
Trans	14
HalfTrans	15

6. Border Colors

Color	Broadcast Value
White	0
Red	1
Green	2
Blue	3

7. Broadcast for Reset

```
{
unsigned char mv0430_output_format;          //output resolution
unsigned char mv0430_border_color;           //border color
unsigned char mv0430_output_layout;          //output layout
unsigned char mv0430_border_enable;          //border enable
unsigned char mv0430_umd_font_size;          //UMD character size
unsigned char reserved_total[5];             //5 reserved, to be expanded
char custom_name[17];                        //custom name (no length in front)
UMD_TOTAL_DATA   m_strcut_umd_data[4];       //page umd
AUDIO_TOTAL_DATA   m_strcut_audio_data[4];   //page audio
OSD_TOTAL_DATA   m_strcut_osd_data[4];       //page osd
}ONE_BTN_SEND_MSG;
```

Examples in hex

Locating a Switcher on the Network

Method: UDP Broadcast
Packet Format: a5 6c 14 00 81 ff 01 00 00 00 00 00 00 00 00 00 ff a5 03 ae
Destination Address: Broadcast 255.255.255.255
Destination Port: 7000

Response Payload:
a5 6c 22 00 a3 ff 01 00 00 00 00 00 00 00 00 00 ff 00 4d 56 30 34 33 30 2d 1b 2d 43 05 30 33 5f 06 ae

Read All the Data of the Device's Current Status
Broadcast
a5 6c 14 00 a3 ff 01 00 00 00 00 00 00 00 00 00 90 58 03 ae

Response Payload
a5 6c 02 01 a3 ff 01 00 00 00 00 00 00 00 00 00 90 00 0a 1b 00 03 00 01 00 00 00 00 00 00 4d 56 30 34 33 30 00 00 00 00 00 00 00 00 00 00 01 01 07 0f 00 00 00 0a 53 00 44 00 49 00 20 00 31 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 01 07 0f 00 00 00 0a 53 00 44 00 49 00 20 00 32 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0101 07 0f 00 00 00 0a 53 00 44 00 49 00 20 00 33 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 01 07 0f 00 00 00 0a 53 00 44 00 49 00 20 00 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 01 00 00 00 00 00 01 00 00 00 00 00 01 00 00 00 00 00 01 00 07 0f 00 00 01 00 07 0f 00 00 01 00 07 0f 00 00 01 00 07 0f 00 00 8c 0a ae

a5 6c 02 01 a3 ff 01 00 00 00 00 00 00 00 00 00	Header format, refer to the previous data structure	
90	Read the command 0x90	
00	0x00 response success	
0a 1b 00 03 00 01 00 00 00 00 00 00	First 12 bytes of ALL_MV0430_CHILD_DLG_DATA	
4d 56 30 34 33 30 00 00 00 00 00 00 00 00 00 00	Device name	
01 01 07 0f 00 00 00	Win 1 UMD parameter	
0a	Win 1 UMD length	
53 00 44 00 49 00 20 00 31 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	Win 1 UMD text character is "SDI 1"	
01 01 07 0f 00 00 00	Win 2 UMD parameter	
0a	Win 2 UMD length	
53 00 44 00 49 00 20 00 32 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	Win 2 UMD text character is "SDI 2"	
0101 07 0f 00 00 00	Win 3 UMD parameter	
0a	Win 3 UMD length	
53 00 44 00 49 00 20 00 33 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	Win 3 UMD text character is "SDI 3"	
01 01 07 0f 00 00 00	Win 4 UMD parameter	
0a	Win 3 UMD length	
53 00 44 00 49 00 20 00 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	Win 4 UMD text character is "SDI 4"	
01 00 00 00 00 00	Win 1	AUDIO parameter, refer to structure AUDIO_TOTAL_DATA
01 00 00 00 00 00	Win 2	
01 00 00 00 00 00	Win 3	
01 00 00 00 00 00	Win4	
01 00 07 0f 00 00	Win 1	OSD parameter, refer to structure OSD_TOTAL_DATA
01 00 07 0f 00 00	Win 2	
01 00 07 0f 00 00	Win 3	
01 00 07 0f 00 00	Win 4	
8c 0a ae	Checksum and package end	

Note: The Character strings here use the ALL_MV0430_CHILD_DLG_DATA from Item 1. to extract the data one by one according to the parameters.

Rename the Device
E.g.： Rename the device to “MV0430”.
Broadcast
a5 6c 1b 00 a3 ff 01 00 00 00 00 00 00 00 00 00 91 4d 56 30 34 33 30 00 ca 04 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 91 00 5a 03 ae

Output Format Setting
E.g.: Setting the output resolution to 1080p50.
Broadcast
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 92 01 5c 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 92 00 5b 03 ae

Border Enable
E.g.: Turn on the border.
Broadcast
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a4 01 6e 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a4 00 6d 03 ae

Set Border Color
E.g.: Set the border color as red.
Broadcast
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 93 01 5d 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 93 00 5c 03 ae

Change Layout Mode
E.g.: Change layout mode as 4-view 3 small windows top and 1 large bottom.
Broadcast
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 94 01 5e 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 94 00 5d 03 ae

UMD Text Enable
E.g.: Turn on the UMD on window 1
Broadcast
a5 6c 16 00 a3 ff 01 00 00 00 00 00 00 00 00 00 95 00 01 60 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 95 00 5e 03 ae

Change UMD Position
E.g.: Change the UMD position to the left of window 1.
Broadcast
a5 6c 16 00 a3 ff 01 00 00 00 00 00 00 00 00 00 96 00 00 60 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 96 00 5f 03 ae

Change UMD Text Color
E.g.: Change UMD color to red on window 1
Broadcast
a5 6c 16 00 a3 ff 01 00 00 00 00 00 00 00 00 00 97 00 02 63 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 97 00 60 03 ae

Change UMD Background Color
E.g.: Change UMD background color to Megenta in window 1.
Broadcast
a5 6c 16 00 a3 ff 01 00 00 00 00 00 00 00 00 00 98 00 03 65 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 98 00 61 03 ae

Set UMD text
E.g.: Set UMD text of window 1 to "SDI 01AAA" (1 space in the middle)
Broadcast
a5 6c 25 00 a3 ff 01 00 00 00 00 00 00 00 00 00 99 00 53 00 44 00 49 00 20 00 31 00 41 00 41 00 41 00 66 05 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 99 00 62 03 ae

Audio Meter Enable
E.g.: Turn on audio meter on window 1.
Broadcast
a5 6c 16 00 a3 ff 01 00 00 00 00 00 00 00 00 00 9a 00 01 65 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 9a 00 63 03 ae

Set Audio Meter Position
E.g.: Change the position of audio meter to right on window 1.
Broadcast
a5 6c 16 00 a3 ff 01 00 00 00 00 00 00 00 00 00 9b 00 02 67 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 9b 00 64 03 ae

Set Audio Source
E.g.: Set audio source of win 1 to CH 7-8
Broadcast
a5 6c 16 00 a3 ff 01 00 00 00 00 00 00 00 00 00 9e 00 03 6b 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 9e 00 67 03 ae

OSD Overlay Enable
E.g.: Turn on OSD on window 1.
Broadcast
a5 6c 16 00 a3 ff 01 00 00 00 00 00 00 00 00 00 9f 00 01 6a 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 9f 00 68 03 ae

Set OSD Position
E.g.: Set OSD to the left on window 1.
Broadcast
a5 6c 16 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a2 00 00 6c 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a2 00 6b 03 ae

Set OSD Text Color
E.g.: setting OSD text color to red on window 1.
Broadcast
a5 6c 16 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a0 00 03 6d 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a0 00 69 03 ae

Set OSD Background Color
E.g.: setting OSD background to VioletRed in Window 1.
Broadcast
a5 6c 16 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a1 00 09 74 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a1 00 6a 03 ae

Set Output Character Size
E.g.: Set output character of all overlays to “middle” size.
Broadcast
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a5 01 6f 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a5 00 6e 03 ae

Save Current Settings to Custom
E.g.: Save the current settings to custom 1.
Broadcast
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a3 01 6d 03 ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a3 00 6c 03 ae

Reset All Settings to Factory Settings
Broadcast
a5 6c ff 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a6 00 00 00 00 00 00 00 00 00 00 4d 56 30 34 33 30 00 00 00 00 00 00 00 00 00 00 00 01 01 07 0f 00 00 00 0a 53 00 44 00 49 00 20 00 31 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 01 07 0f 00 00 00 0a 53 00 44 00 49 00 20 00 32 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 01 07 0f 00 00 00 0a 53 00 44 00 49 00 20 00 33 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 01 07 0f 00 00 00 0a 53 00 44 00 49 00 20 00 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 01 00 00 00 00 00 01 00 00 00 00 00 01 00 00 00 00 00 01 00 07 0f 00 00 01 00 07 0f 00 00 01 00 07 0f 00 00 01 00 07 0f 00 00 75 0b ae

Broadcast Description

a5 6c ff 00 a3 ff 01 00 00 00 00 00 00 00 00 00	Header format, refer to the previous data structure	
a6	0xa6，reset command	
00 00 00 00 00 00 00 00 00 00	ONE_BTN_SEND_MSG First 10 bytes	
4d 56 30 34 33 30 00 00 00 00 00 00 00 00 00 00	Device name	
01 01 07 0f 00 00 00	win 1 UMD parameter	
0a	win 1 UMD length	
53 00 44 00 49 00 20 00 31 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	win 1 UMD text Character is “SDI 1”	
01 01 07 0f 00 00 00	win 2 UMD parameter	
0a	win 2 UMD length	
53 00 44 00 49 00 20 00 32 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	win 2 UMD text Character is “SDI 2”	
01 01 07 0f 00 00 00	win 3 UMD parameter	
0a	win 3 UMD length	
53 00 44 00 49 00 20 00 33 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	win 3 UMD text Character is “SDI 3”	
01 01 07 0f 00 00 00	win 4 UMD parameter	
0a	win 4 UMD length	
53 00 44 00 49 00 20 00 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	win 4 UMD text Character is “SDI 4”	
01 00 00 00 00 00	Win 1	AUDIO parameter, refer to structure AUDIO_TOTAL_DATA
01 00 00 00 00 00	Win 2	
01 00 00 00 00 00	Win 3	
01 00 00 00 00 00	Win 4	
01 00 07 0f 00 00	Win 1	OSD parameter, refer to structure OSD_TOTAL_DATA
01 00 07 0f 00 00	Win 2	
01 00 07 0f 00 00	Win 3	
01 00 07 0f 00 00	Win 4	
75 0b ae	checksum and package end	

Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a6 00 6f 03 ae

Noted: Character string pack as ONE_SEND_MSG, please refer to structure of 4.7 part.

5.23 IP Address Settings
E.g.: Set IP address: 192.168.1.234
Subnet mask: 255.255.255.0
Default gateway: 192.168.1.1
Broadcast
a5 6c 20 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a7 c0 a8 01 ea ff ff ff 00 c0 a8 01 01 35 0a ae
Response Payload
a5 6c 15 00 a3 ff 01 00 00 00 00 00 00 00 00 00 a7 00 70 03 ae

Sample C# Application to locate MVS-3 and change properties

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Net.Sockets;
using System.Net;
using System.Globalization;

namespace OspreyMultiViewerAPI
{
    class Program
    {
        static void Main(string[] args)
        {
            Sender s = new OspreyMultiViewerAPI.Sender();
            s.Send();
        }
    }
    public class Sender
    {

        public void Send()
        {
            UdpClient client = new UdpClient();

     // First thing we are going to do is locate the Multiviewer.  The command is 0xff for scan

            client.EnableBroadcast = true;
            IPEndPoint broadcastConnAddress = new IPEndPoint(IPAddress.Broadcast, 7000);
            byte[] bytes = HexToByte("a56c140081ff010000000000000000ffa503ae");
            client.Send(bytes, bytes.Length, broadcastConnAddress);
            IPEndPoint ServerEp = new IPEndPoint(IPAddress.Any, 0);
     // Wait for a response
            var ServerResponseData = client.Receive(ref ServerEp);
            Byte type = ((byte[])ServerResponseData)[4];
     // A success bit of 0 indicates data returned successfully.
            Byte success = ((byte[])ServerResponseData)[17];
            bool bSuccess = false;
            if (success == 0)
                bSuccess = true;
```

```
            Console.WriteLine(@"Response from with IP address: {0} with type: {1} and success
            of: {2}",
            ServerEp.Address.ToString(),
            String.Format("{0:x2}", type), bSuccess.ToString());
            /* Now we attempt to setup  the multiviewer.
            * At this point all communication is
            * directed on port 7000 of the matrix swicher's IP address
            * Lets turn on the OSD on window 1 (OSD Overlay Enable)
            */

            bytes = HexToByte("a56c1600a3ff010000000000000000009f00016a03ae");
            client.Send(bytes, bytes.Length, broadcastConnAddress);
            ServerEp = new IPEndPoint(IPAddress.Any, 0);
            // Wait for a response
            ServerResponseData = client.Receive(ref ServerEp);

            // A success bit of 0 indicates data returned successfully.
            success = ((byte[])ServerResponseData)[17];
            bSuccess = false;
            if (success == 0)
                bSuccess = true;


            // Close the connection
            client.Close();
            return;
        }

        public static byte[] HexToByte(string hexString)
        {
            if (hexString.Length % 2 != 0)
            {
                throw new ArgumentException(String.Format(CultureInfo.InvariantCulture,
                "The binary key cannot have an odd number of digits: {0}", hexString));
            }
            byte[] HexAsBytes = new byte[hexString.Length / 2];
            for (int index = 0; index < HexAsBytes.Length; index++)
            {
                string byteValue = hexString.Substring(index * 2, 2);
                HexAsBytes[index] = byte.Parse(byteValue, NumberStyles.HexNumber,
                CultureInfo.InvariantCulture);
            }
            return HexAsBytes;
        }
    }
}
```