VA-2002 RS232 Remote Interface Specification

V2.1 Nov.2003

This document describes the communication protocol between the VA-2002 device and an host computer, for example a PC. A serial RS232 interface is used to talk with the device. The parameters are 8 data bits, no stop bits, no handshake, 115200 Baud (version <2.0 uses 38400 Baud).

Communication Protocol

Every communication is initiated by the PC. To keep up-to-date with the state of the device, the PC should issue a BOX_STATUS command periodically.

The format of each communication is as follows:

PC command (with CRC protection)

Byte	Meaning
CMD	Command to be executed, see list below (numbers 0x00 to 0x7F are allowed)
Par 1	First parameter for this command
…	…
Par n	Last parameter for this command
CRC Low	16 Bit CRC low byte
CRC High	16 Bit CRC high byte
0xFE	Executes command

There is an alternative way to send a command to the device, which is not CRC protected (not recommended

PC command (without CRC protection)

Byte	Meaning
CMD	Command to be executed, see list below (numbers 0x00 to 0x7F are allowed)
Par 1	First parameter for this command
…	…
Par n	Last parameter for this command
0xFF	Executes command

Because 0xFE and 0xFF can't be sent as Par or CRC, there is a translation for these bytes:

Byte translation

Byte	Translates to
0xA5	0xA5 0xA5
0xA6	0xA5 0xA6
0xA7	0xA5 0xA7
0xFE	0xA6
0xFF	0xA7

The device executes the command, and replies with the following sequence :

Device reply

Byte	Meaning
CMD | 0x80	Or'ed byte to indicate that answer fits to command
Par 1	First reply
…	…
Par n	Last reply
0xFF	End mark

If the device isn't able to execute the command, if replies by sending BUSY:

Device reply (device is busy):

Byte	Meaning
0xFE	Device is busy

Please notice that logos are numbered 0-8 in this document, although the user-interface displays the numbers starting with 1!

The following pages describe the commands in detail.

Commands

BOX_INITINFO

Command: 0x00
Parameters: NONE
Reply: 12 Byte

Reply is initialisation Information:

Byte	Name	Description
2 Byte	TOTAL_MEM	Total memory of the VA-2002
1 Byte	OS_VER_MAIN	OS Version Main (1)
1 Byte	OS_VER_SUB	OS Version Sub (1)
4 Byte	SERIAL_NUM	Serial number (2)
4 Byte	DEVTYPE_NUM	Identifies device (3)

Description:

This command receives fixed information about the device. This command does only need to be issued once after starting the program, or when (re)connecting the device.

Remarks:

- Current OS Version is: V2.1
- The serial number is fixed to 0x00000000, it is reserved for future use
- Device Type is 1 for VA-2002, you should abort your program if you see a different device.

BOX_STATUS

Command: 0x01
Parameters: NONE
Reply: 1 Byte

Reply is status Information:

Byte	Name	Description
1 Byte	STATUS	Status Info, see bit description below

Bit defintion of STATUS byte:

Bit	Name	Description
Bit 1-0=0	VISIBLE_STATE	Logo is OFF
Bit 1-0=1		Logo is ON
Bit 1-0=2		Logo is FADING OFF->ON
Bit 1-0=3		Logo is FADING ON->OFF
Bit 2	VIDEO_LOCKED	Video Locked Flag (1=locked)
Bit 3	NOTIFY_LOGO	1=notification, that list of logos has changed (1)
Bit 4	NOTIFY_PREFS	1=notification, that preferences have changed (2)
Bit 5	CRITICAL_MENU	1=user is in critical menu (3)
Bit 6	ENABLED_SDI	1=SDI feature is installed

Description:

This command returns only 1 byte to describe the status of the device. If one or more of the NOTIFY_xxx bits are set, additional information about the status change should be obtained by calling the appropriate commands. It should be called periodically to be keeped informed about the status of the device.

Remarks:

(1)If NOTIFY_LOGO is set, the command BOX_LOGO should be issued to obtain the actual logo-list.

(2)If NOTIFY_PREFS is set, the command BOX_PREFS should be issued to obtain the actual preferences.

(3)For CRITICAL_MENU, see chapter “The problem of concurrent input” below.

BOX_LOGO

Command: 0x02
Parameters: NONE
Reply: 147Byte

Reply is logo Information:

Byte	Name	Description
1 Byte	NUM_LOGOS	Number of installed logos (1)
1 Byte	ACT_LOGO	Number of actual selected logo
1 Byte	USED_MEM	Amount of used memory in percent
16 Byte	LOGO_NAME_0	Name of logo 0 (2)
…		
16 Byte	LOGO_NAME_8	Name of logo 8 (2)

Description:

This command returns the list of all logos. It should be executed after startup to obtain the list, and each time, when NOTIFY_LOGO is set in the BOX_STATUS command reply. Please notice that logos are numbered starting with 0.

Remarks:

(1)LOGO_NAME_n for n>=NUM_LOGOS contains no valid data

(2)The name is zero-terminated, as long as the length is less than 16. If the length is 16, you need to add the terminating 0 by yourself.
The sequence of the characters for each logo is 3-2-1-0-7-6-5-4-11-10-9-8-15-14-13-12, because the device processor is 32bit Big-endian. In order to bring the characters in the correct order, execute the following loop:
for (i=0; i<16; i++) logo[i] = rx[i xor 3]; logo[16]=0;

BOX_PREFS

Command: 0x03
Parameters: NONE
Reply: 5 Byte

Reply is preferences Information:

Byte	Name	Description
1 Byte	INPUT	Source of input: 0=composite (BNC) 1=Y/C (mini DIN) 2=SDI (BNC)
1 Byte	STANDARD	Type of video standard used: 0=PAL 1=NTSC
1 Byte	MONITOR	Type of picture displayed on monitor output: 0=used as control 1=used as monitor
1 Byte	UNUSED	This byte contains no valid information
1 Byte	LOCK	Locked state of the device: 0=device is unlocked 1=device is locked

Description:

This command returns the Preferences of the device.

BOX_SHOW

Command: 0x04
Parameters: NONE
Reply: NONE

Description:

To switch on/off the logo, use this command. Please notice, that you only can switch the logo from one state to the other (it simulated the press on the SWITCH button). To switch the logo to a desired state (e.g. ON), first call the BOX_STATUS command and examine the VISIBLE_STATE bits, and only issue this command, if the logo isn't already in the desired state.

BOX_SETINPUT

Command: 0x05
Parameters: 1 Byte
Reply: NONE

Parameter is desired input:

Byte	Name	Description
1 Byte	INPUT_SELECT	Source of input to activate: 0=composite (BNC) 1=Y/C (mini DIN) 2=SDI (BNC)

Description:

This command switches the input to the desired state. Before switching to SDI ensure that the ENABLED_SDI flag is set in the BOX_INITINFO command.

BOX_SETSTANDARD

Command: 0x06
Parameters: 1 Byte
Reply: NONE

Parameter is desired standard:

Byte	Name	Description
1 Byte	STANDARD_SELECT	Type of video standard to activate: 0=PAL 1=NTSC

Description:

This command switches the standard to the desired state.

BOX_SETMONITOR

Command: 0x07
Parameters: 1 Byte
Reply: NONE

Parameter is desired monitor mode:

Byte	Name	Description
1 Byte	MONITOR_SELECT	Type of picture displayed on monitor output: 0=used as control 1=used as monitor

Description:

This command switches the monitor to the desired state.

BOX_SETLOCK

Command: 0x08
Parameters: 1 Byte
Reply: NONE

Parameter is lock on/off:

Byte	Name	Description
1 Byte	LOCK_SELECT	Locked state of the device to activate: 0=unlock device 1=lock device

Description:

This command locks or unlocks the device. If you *lock* the device, you need to issue the command BOX_KICKSTATUSMENU command before.

BOX_GETLOGOINFO

Command: 0x09
Parameters: NONE
Reply: 6 Bytes

Reply is logo information:

Byte	Name	Description
2 Byte	LOGO_MEMORY	Used memory by this logo
2 Byte	LOGO_XSIZE	X-Size in pixels for this logo
2 Byte	LOGO_YSIZE	Y-Size in pixels for this logo

Description:

This command obtains information about one particular logo. Please notice that logos are numbered starting with 0.

BOX_KICKHARMLESSMENU

Command: 0x0A
Parameters: NONE
Reply: NONE

Description:

See chapter “The problem of concurrent input” below.

BOX_DELETELOGO

Command: 0x0B
Parameters: 1 (number of logo)
Reply: NONE

Description:

This command deletes one particular logo. Please notice that logos are numbered starting with 0.

This command only works, if no logo is currently visible on the screen (VISIBLE_STATE is 0 in the BOX_STATUS command)!

BOX_SELECTLOGO

Command: 0x0C
Parameters: 1 (number of logo)
Reply: NONE

Description:

This command selects one particular logo for displaying. Please notice that logos are numbered starting with 0.

BOX_DELETEALLLOGOS

Command: 0x0D
Parameters: NONE
Reply: NONE

Description:

This command deletes all logos.

This command only works, if no logo is currently visible on the screen (VISIBLE_STATE is 0 in the BOX_STATUS command)!

BOX_KICKSTATUSMENU

Command: 0x0E
Parameters: NONE
Reply: NONE

Description:

This command forces the device to go in the status menu. Used together with the BOX_LOCK command (only if the lock is *set*).

BOX_LOADLOGOSTART

Command: 0x0F
Parameters: 252
Reply: 1

Description:

The logo data is transferred in blocks of 252 bytes.

Use this function to transfer the first block. After this block the VA-2002 will check the parameters contained in this block (size, name, etc), and send a reply to the PC.

After successfully sending this blocks, the command BOX_LOADLOGOLOOP has to be used for the remaining blocks. Finally BOX_LOADLOGOEND has to be called.

For description of the file-format see chapter below.

If a logo with the same name is already present in the VA-2002, it will be overwritten without asking. If the user should be informed, the command BOX_LOGO has to be used to find if the logo is present.

The following table gives error messages that will prevent the logo from loading:

Reply from BOX_LOADLOGOxxx

Reply	Name	Description
0	LOADLOGO_OKAY	success
1	LOADLOGO_LOGOVISIBLE	Logo is visible
2	LOADLOGO_TOOMANYLOGOS	Already 9 logos stored
3	LOADLOGO_OUTOFMEM	No memory left
4	LOADLOGO_TIMEOUT	Time between two BOX_LOADLOGOxxx command too long
5	LOADLOGO_TOOMUCHDATA	BOX_LOADLOGOLOOP command called too often
6	LOADLOGO_NOTENOUGHDATA	BOX_LOADLOGOLOOP command not called often enough

BOX_LOADLOGOLOOP

Command: 0x10
Parameters: 252
Reply: 1

Description:

This command is used to load the remaining blocks of the logo. For reply see BOX_LOADLOGOSTART.

BOX_LOADLOGOEND

Command: 0x11
Parameters: NONE
Reply: 1

Description:

This command must be issued after loading the complete logo. It stores the logo permanently in the VA-2002. For reply see BOX_LOADLOGOSTART.

The problem of concurrent input

The device accepts commands via:

- Input from the front panel (*direct*)
- From the RS232 interface (*remote*)

Because the device can receive commands *direct* and *remote* simultaniously, care must be taken not to perform inconsistent actions.

Example:

The program wants to delete logo 4 by *remote*, and asks the user for confirmation on the PC-screen. In the meanwhile the user deletes the first logo *directly*. After that the program continues (it got a positive answer to the confirmation dialog) by executing the BOX_DELETELOGO command on logo 4. Then the device would delete the logo, which was logo 5 before (because all logos are renumbered after deleting the first logo).

“Bad” users even try to execute the delete commands *directly* and *remotely* at the same time (especially users who are much too curious ☺), so how can we prevent this?

Two mechanisms to prevent inconsistency:

The *remote* PC is able to supervise if the user is able to execute a “dangerous” command, by examining the CRITICAL_MENU flag in the BOX_STATUS command.

The *direct* input can be forced by *remote* to leave such a critical menu by executing the BOX_KICKHARMLESSMENU command.

Example revisited:

- The user navigates in to a critical menu on the front panel *directly*.
- The PC *remote* wants to execute a critical command (e.g. Delete Logo).
- The PC *remote* checks the CRITICAL_MENU flag first, and sees that it is set.
- The PC *remote* executes the BOX_KICKHARMLESSMENU command.
- Now the PC *remote* displays the “do you really want to delete this logo?” confirmation dialog.
- If the user *directly* navigates in a critical menu (again), the CRITICAL_MENU flag becomes set again, and the PC *remote* should abort the operation.

You need to issue the previous described sequence only on the following commands:

BOX_DELETELOGO, BOX_DELETEALLLOGOS, BOX_SELECTLOGO, BOX_LOADLOGOSTART

Data format for BOX_LOADLOGOxxx commands

The following data has to be partitioned in 252 byte blocks and sent via the BOX_LOADLOGOxxx commands as described above. Please notice, that video data has to be converted from RGB to “YUYV” format, and transparency data has to be added for each YUYV sequence.

Data to be sent to the VA-2002

Size	Description
1 Long	“LOGO”
1 Short	Version (0x0001)
1 Short	Number of 252 blocks to be transferred
16 Bytes	Name of Logo (see comment (2) of BOX_LOGO command)
1 Short	Len of Header in Longs (0x0003)
1 Short	unused (0x0000)
1 Short	Width (must be multiple of 4)
1 Short	Height
1 Short	X-position
1 Short	Y-position
1 Short	FadeSpeed
1 Short	Destination transparency
n Longs	Data (first all odd lines, then all even lines), see below

For each 4 pixel tuple packet the following sequence will be sent:

Size	Name	Description
1 Byte	TA	Transparency of pixels tuple A
1 Byte	TB	Transparency of pixels tuple B
1 Byte	TC	Transparency of pixels tuple C
1 Byte	TD	Transparency of pixels tuple D
1 Long	PIXA	Pixel tuple A Data (only, if TA and/or TB is not zero)
1 Long	PIXB	Pixel tuple B Data (only, if TA and/or TB is not zero)
1 Long	PIXC	Pixel tuple C Data (only, if TC and/or TD is not zero)
1 Long	PIXD	Pixel tuple D Data (only, if TC and/or TD is not zero)

Pixel tuple data is as follows:

Size	Description
1 Byte	Y-left
1 Byte	U
1 Byte	Y-right
1 Byte	V

Project file data format

Video data is stored in ARGB format (A is the alpha channel). Before sending via RS232 to the VA-2002, video data has to be converted in YUYV format, and the header has to be changed.

Size	Description
1 Long	“LPRJ”
1 Long	Version
1 Long	X-position
1 Long	Y-position
1 Long	Width
1 Long	Height
1 Long	FadeSpeed (leave out for Version 1)
1 Long	unused (leave out for Version 1)
Height*Width Long	Data (starting with last line!)

For each pixel a Long is stored, which contains the following data:

Size	Description
1 Byte	blue
1 Byte	green
1 Byte	red
1 Byte	transparency