

VCON

Group Videoconferencing Systems



HDK Integrator's Commands

Version 2.0

Programmer's Guide

© 2003, 2004 VCON Ltd. All Rights Reserved.

This material is proprietary of VCON, Ltd. No part of this document can be reproduced or transmitted in any form or by any means - electronic or mechanical - for any purpose without written permission from VCON Ltd. Information in this document is subject to change without notice.

Limited Warranty

VCON Ltd. warrants that SOFTWARE will perform according to accompanying user documentation for a period of 90 (ninety) days from the date of receipt; replacement SOFTWARE will be warranted for 90 (ninety) days from the date of receipt. This Limited Warranty shall not apply to any product that in the opinion of VCON Ltd. has not been installed or upgraded according to accompanying documentation from VCON Ltd. or been subject to misuse, misapplication, negligence or accident while in the hands of the purchaser.

GRANT OF LICENCE VCON Ltd. grants the Purchaser a non-exclusive and non-transferable license to use the SOFTWARE product and to make one copy solely for backup or archival purposes, which may include user documentation provided via online or other electronic form. Additional copies may not be made nor may anyone else be allowed to copy or otherwise reproduce any part of the licensed software without prior written consent of VCON Ltd.

COPYRIGHT All trademarks(s), logo(s), name(s), software, documentation and other supporting materials relating to the Product are trademarked, copyrighted or owned by VCON Ltd. as proprietary information protected by United States copyright laws and international and applicable national treaty provisions and laws. Software protection extends beyond its literal code to structure, sequence and organization; any unauthorized use or modification would constitute a misappropriation of VCON's proprietary rights and a violation of the License agreement.

LIABILITIES VCON's entire liability and the Purchaser's exclusive remedy shall be at VCON's option, either return of the price paid or repair/replacement of the Product not meeting VCON's declared Limited warranty. VCON or its suppliers shall not be liable in any event to anyone for any indirect, incidental, consequential, special or exemplary damages including without limitation damages for loss of business profits, business interruptions, business information or other pecuniary loss arising out of the use of or inability to use the said Product even if advised of the possibility of such damages. In any case, VCON's entire liability under any provision of this agreement shall be limited to the amount actually paid by the Purchase for the Product.

VCON Technical Support

This Programmer's Guide was designed to help you integrate HD systems with your hardware. It contains descriptions, syntaxes, parameters, and examples of the HDK commands and events.

If a situation occurs that is not covered by the supplied documentation, please request help from our Technical Support channels. VCON's organization will make its strongest efforts to help you resume your software integration activities as soon as possible.

1. Contact your local VCON distributor, and request assistance from its technical support department.
2. Send an e-mail message fully describing the condition plus your system's configuration to zapi@vcon.com.

TABLE OF CONTENTS

Limited Warranty.....	2
VCON Technical Support.....	3
1 Accessing the HD Programming Interface.....	10
1.1 Setting Up an RS-232 Connection.....	10
1.2 Setting Up a Telnet Session.....	11
Hiding the HD User Interface.....	12
2 Syntax Conventions.....	13
3 Commands.....	14
3.1 Call Commands.....	14
Dial.....	14
Hangup.....	15
CallAnswer.....	16
CallReject.....	17
CallGetParam.....	18
STATE.....	20
REMOTE_TERMINAL_TYPE.....	21
DIRECTION.....	22
BITRATE.....	23
TX_VIDEO_CODEC.....	24
RX_VIDEO_CODEC.....	25
TX_VIDEO_SEC_CODEC.....	26
RX_VIDEO_SEC_CODEC.....	27
TX_AUDIO_CODEC.....	28
RX_AUDIO_CODEC.....	29
TX_VIDEO_BITRATE.....	30
RX_VIDEO_BITRATE.....	31
TX_VIDEO_SEC_BITRATE.....	32
RX_VIDEO_SEC_BITRATE.....	33
TX_AUDIO_BITRATE.....	34
RX_AUDIO_BITRATE.....	35
TX_VIDEO_HANDLE.....	36
RX_VIDEO_HANDLE.....	37
TX_VIDEO_SEC_HANDLE.....	38
RX_VIDEO_SEC_HANDLE.....	39
TX_AUDIO_HANDLE.....	40
RX_AUDIO_HANDLE.....	41

TX_VIDEO_FRAMERATE	42
RX_VIDEO_FRAMERATE	43
TX_VIDEO_SEC_FRAMERATE	44
RX_VIDEO_SEC_FRAMERATE	45
DURATION	46
REMOTE_NAME	47
REMOTE_ADDRESS	48
AUDIO_TO_VIDEO_DELAY	49
JITTER_BUFFER_SIZE	50
TX_VIDEO_RESOLUTION	51
RX_VIDEO_RESOLUTION	52
TX_VIDEO_SEC_RESOLUTION	53
RX_VIDEO_SEC_RESOLUTION	54
CallSetBitrate	55
CallSendIntra	56
CallRequestIntra	57
CallSendDTMF	58
CallEvents	59
3.2 LAN Commands	60
GetParam	60
GATEKEEPER_LOGIN_STATE	63
GATEKEEPER_ADDRESS	64
STATION_NUMBER	65
STATION_NAME	66
SERVER_LOGIN_STATE	67
SERVER_ADDRESS	68
AUTO_ADDRESS	69
MAC_ADDRESS	70
IP_ADDRESS	71
SUBNET_MASK	72
DEFAULT_GATEWAY	73
DNS_SERVER_ADDRESS	74
WINS_SERVER_ADDRESS	75
DOMAIN_NAME	76
NAT_ADDRESS	77
AUTO_ANSWER	78
LIPSYNC_ENABLE	79
AUTO_JITTER_ENABLE	80
ABA_ENABLE	81
MAX_BITRATE	82
MIN_BITRATE	83

DEF_BITRATE	84
USE_GATEKEEPER	85
USE_SERVER	86
MIN_UDP_PORT	87
MAX_UDP_PORT	88
MIN_TCP_PORT	89
MAX_TCP_PORT	90
AUDIO_QOS	91
VIDEO_QOS	92
RTCP_QOS	93
QOS_TYPE	94
AUTO_ACCEPT_MULTICAST_FLOOR	95
DHCP_STATE	96
WINS_LOGIN_STATE	97
STREAMING_ENABLE	98
STREAMING_ADDRESS	99
STREAMING_VIDEO_PORT	100
STREAMING_AUDIO_PORT	101
STREAMING_BITRATE	102
STREAMING_TTL	103
STREAMING_TIMEOUT	104
STREAMING_ENABLE_ANNOUNCEMENT	105
ETHERNET_NIC_STATUS	106
ETHERNET_NIC_SETTINGS	107
MAX_ACTIVE_CALLS	108
ACTIVE_CALLS	109
FECC_ALLOWED	110
ANSWER_MODE_P2P	111
ANSWER_MODE_MULTIPOINT	112
BAN_OUTGOING_CALLS	113
CRYPTO_ALGORITHM	114
SetParam	115
ServerEvents	116
GatekeeperEvents	117
3.3 Streaming Commands	118
StreamingStart	118
StreamingStop	119
StreamingResume	120
StreamingEvents	121
3.4 Interactive Multicast Commands	122
MulticastSendPassword	122

	MulticastSendText	123
	MulticastRequestFloor	124
	MulticastRejectFloor	125
	MulticastAcceptFloor	126
	MulticastEvents	127
3.5	Camera Commands	128
	CameraGetSelected	128
	CameraSetSelected	129
	CameraPresetSet	130
	CameraPresetRecall	131
	CameraMove	132
3.6	Video Commands	133
	MuteVideo	133
	DisplaySet	134
3.7	Audio Commands	135
	AudioSetMicInput	135
	AudioGetMicInput	136
	AudioSetMicGainLevel	137
	AudioGetMicGainLevel	138
	AudioFilePlay	139
	AudioFileStop	140
	MuteMic	141
	MuteSpeaker	142
	VolumeGet	143
	VolumeSet	144
3.8	Phonebook Commands	145
	PhonebookGetEntry	145
	PhonebookGetNumOfEntries	146
	PhonebookAddEntry	147
	PhonebookDeleteEntry	148
	PhonebookShow	149
	PhonebookDial	150
3.9	System Commands	151
	IniRead	151
	IniWrite	152
	Reboot	153
	KeyPressed	154
	ConsoleGetParam	156
	ConsoleSetParam	157
	DiskInfo	158
	Ping	159

ShowDbgMsg	160
4 Events	161
4.1 Call Events	161
EVENT_CALL_DIALTONE	161
EVENT_CALL_RINGBACK	162
EVENT_CALL_CONNECTED	163
EVENT_CALL_OFFERING	164
EVENT_CALL_DISCONNECTED	165
EVENT_CALL_IDLE	166
EVENT_CALL_CHANNEL_ON	167
EVENT_CALL_CHANNEL_OFF	168
EVENT_CALL_REMOTE_NOT_RESPONDING	169
4.2 Server Events	170
EVENT_SERVER_DATA_CHANGED	170
EVENT_SERVER_COMMAND	171
EVENT_SERVER_STATE_CHANGED	172
4.3 Gatekeeper Events	173
EVENT_GATEKEEPER_STATE_CHANGED	173
4.4 Multicast Events	174
EVENT_MULTICAST_CONNECTED	174
EVENT_MULTICAST_FLOOR_OFFERING	175
EVENT_MULTICAST_FLOOR_REQUEST_REJECTED	176
EVENT_MULTICAST_FLOOR_CHANGED	177
EVENT_MULTICAST_FLOOR_GRANTED	178
EVENT_MULTICAST_PASSWORD_REJECTED	179
EVENT_MULTICAST_PASSWORD_REQUESTED	180
4.5 Streaming Events	181
EVENT_STREAMING_STARTED	181
EVENT_STREAMING_STOPPED	182
4.6 System Events	183
EVENT_SYSTEM_INIT_COMPLETED_SUCCESS	183
EVENT_SYSTEM_INIT_COMPLETED_FAILURE	184
EVENT_SYSTEM_DHCP_STATE_RECEIVED_ADDRESS	185
EVENT_DHCP_STATE_FAILED_RECEIVING_ADDRESS	186
4.7 FECC Events	187
CAMERA_REMOTE_MOVE	187
CAMERA_LOCAL_MOVE	188
5 The User.ini File	189
5.1 Configuration Parameters	190
CallEvents	190

	MulticastEvents	191
	StreamingEvents	192
	GatekeeperEvents	193
	ServerEvents	194
5.2	GUI Parameters.....	195
	HideAll	195
	HideDlg	196
6	WorkFlow Examples	197
6.1	Incoming Call Flow	197
6.2	Outgoing Call Flow	199

1 **ACCESSING THE HD PROGRAMMING INTERFACE**

The HDK (High-definition Development Kit) is a set of APIs used by VCON's HD100, HD2000, and HD3000 to build custom applications (such as distance learning, security, or kiosks). The HDK may be implemented as a PC program, AMX, Creston, or any other programmable system.

1.1 **Setting Up an RS-232 Connection**

This section includes procedures for setting up an RS-232 communications session.

➤ **To connect a PC to the HD device**

- Connect one end of the specified communication cable to a COM port of your computer and to a connector on the HD device as follows:

HD100 Connect CAB42001 communication cable to the HD100's COM2 port.

HD2000/HD3000 Connect CAB00026 serial-to-dual cable to the HD device's RS232 connector, and the cable end labelled **User Terminal** to one of the PC COM ports.

➤ **To start a communications session**

1. Set up a HyperTerminal session. In the Windows desktop, run **Start>Programs>Accessories>Communications>HyperTerminal**.
2. In the Connection Description dialog box, enter a **Name**. Select an icon. Click **OK**.
3. In the Connect To dialog box, select **COM1** from the **Connect Using** list. Click **OK**.
4. In the COM Properties dialog box, enter the following Port Settings:

Bits per second 9600

Data Bits 8

Parity None

Stop Bit 1

Flow Control None

5. Click **OK**.
6. Restart the HD device. After it initializes, the following prompt appears on the screen:

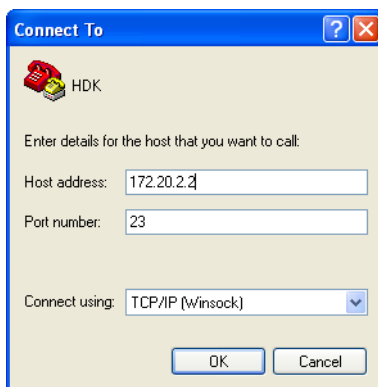
```
Welcome to HD3000:  
Version      : 0200.M03.D03.H19  
Board Type   : HD3000 Board  
Camera Input: NTSC  
  
HD>>  
EVENT_SYSTEM_INIT_COMPLETED_SUCCESS  
HD>>
```

1.2 *Setting Up a Telnet Session*

Use standard Telnet client configured with the HD device's IP address and **port 23**.

► **To start a Telnet session from Windows**

1. Turn the HD device on.
2. From the Windows **Start** menu, open **Programs>Accessories>Communications>HyperTerminal**.
3. Enter a session.



4. From the **Connect Using** list, choose **TCP/IP (Winsock)**.

5. Enter the HD device's IP address.
6. As the **Port Number**, type **23**.
7. Click **OK**.

You are now in an active Telnet session. Information from the HD device appears on the screen.

```
Welcome to HD3000:
Version      : 0200.M03.D03.H19
Board Type   : HD3000 Board
Camera Input: NTSC

HD>>
EVENT_SYSTEM_INIT_COMPLETED_SUCCESS
HD>>
```



To display the list of available API commands, type **help**.

Hiding the HD User Interface

If you want to remove the entire user interface or only its dialog boxes from view, perform the procedures below.

► To hide the HD user interface

1. Connect to the HD device over Telnet (see [“Setting Up a Telnet Session” on page 11](#)).
2. In the Telnet session screen, type **IniWrite user.ini gui hideall 1**.

► To hide only dialog boxes (icons and popup messages remain visible):

1. Connect to the HD device over Telnet (see [“Setting Up a Telnet Session” on page 11](#)).
2. In the Telnet session screen, type **IniWrite user.ini gui hiddenlg 1**.

2 SYNTAX CONVENTIONS

The following conventions are used for the API command syntaxes:

- Parameters placed between < > are mandatory.
- Parameters placed between [] are optional.
- Only one of the parameters placed between { } and separated by | may be used at a single time.
- Available values appear in **bold**.
- Default values appear in ***bold italics***.
- Words or characters appearing in *italics* represent variable values that you must supply.
- Examples of text that appear on the monitor (application console) appear in `Courier` font.
- The return string for all commands is structured as follows:
 - `<CommandName> <Status={OK|ERROR}> [Value]`
- Links to other events, commands, and parameters appear in [blue](#).



Commands are not case sensitive.

3 COMMANDS

This chapter contains description, syntaxes, parameters, and examples of the HDK commands.

3.1 Call Commands

Dial

Description	Initiate a videoconference call to a remote user associated with the entered address.
Format	Dial < <i>RemoteAddress</i> > [<i>Bitrate</i> = 384000]
Parameters	RemoteAddress – The address of the remote user (IP Address, DNS Name H.323-ID or E.164). Bitrate – The bitrate of the call.
Return	OK on success together with the created call ID. ERROR – with an error code.
Example	<pre>Dial 10.0.11.25 128000 DIAL OK 0 Dial danny.vcon.co.il 128000 DIAL OK 1</pre>
Remarks	

Hangup

Description	Disconnect an active call.
Format	Hangup [<i>CallID=0</i>]
Parameters	CallID – The ID of the call to disconnect.
Return	OK on success. ERROR – with an error code.
Example	Hangup 0 HANGUP OK
Remarks	

CallAnswer

Description	Answer an incoming call.
Format	CallAnswer [<i>CallID=0</i>]
Parameters	CallID – The ID of the incoming call.
Return	OK on success. ERROR – with an error code.
Example	CallAnswer 0 CALLANSWER OK
Remarks	When the system is set to auto-answer mode, this command is unnecessary.

CallReject

Description	Reject an incoming call.
Format	CallReject [<i>CallID=0</i>]
Parameters	CallID – The ID of the incoming call.
Return	OK on success. ERROR – with an error code.
Example	CallReject 0 REJECTCALL OK
Remarks	When the system is set to auto-answer mode, this command is unnecessary.

CallGetParam

Description	Retrieve a call parameter.
Format	CallGetParam [<i>CallID=0</i>] < <i>CallParameter</i> >
Parameters	CallID – The ID of the call. CallParameter – The specific call parameter.
Return	OK on success together with the parameter value. ERROR on error together with an error code.
Example	<pre>CallGetParam 0 REMOTE_NAME CALLGETPARAM OK DemoRoom</pre>
Remarks	To retrieve all the Call parameters, type GetCallParam . NULL string represents a null value. Supported call parameters list: STATE REMOTE_TERMINAL_TYPE DIRECTION BITRATE TX_VIDEO_CODEC RX_VIDEO_CODEC TX_VIDEO_SEC_CODEC RX_VIDEO_SEC_CODEC TX_AUDIO_CODEC RX_AUDIO_CODEC TX_VIDEO_BITRATE RX_VIDEO_BITRATE TX_VIDEO_SEC_BITRATE RX_VIDEO_SEC_BITRATE

**Remarks
(cont.)**

TX_AUDIO_BITRATE
RX_AUDIO_BITRATE
TX_VIDEO_HANDLE
RX_VIDEO_HANDLE
TX_VIDEO_SEC_HANDLE
RX_VIDEO_SEC_HANDLE
TX_AUDIO_HANDLE
RX_AUDIO_HANDLE
TX_VIDEO_FRAMERATE
RX_VIDEO_FRAMERATE
TX_VIDEO_SEC_FRAMERATE
RX_VIDEO_SEC_FRAMERATE
DURATION
REMOTE_NAME
REMOTE_ADDRESS
AUDIO_TO_VIDEO_DELAY
JITTER_BUFFER_SIZE
TX_VIDEO_RESOLUTION
RX_VIDEO_RESOLUTION
TX_VIDEO_SEC_RESOLUTION
RX_VIDEO_SEC_RESOLUTION

Descriptions for the parameters listed above appear on the following pages.

STATE

Description	Start or stop sending call event to the console. Type of call event to the console. Current call state.
Valid Value	DIALTONE - A dialtone state followed dialing. DISCONNECTED - A call was disconnected. RINGBACK - An “Alerting” or “Proceeding” message was received from the remote side. CONNECTED - A call connected successfully. IDLE - A call was disconnected, and its resources are no longer valid. OFFERING - An incoming call indication was received.
Access	Read only.
Remarks	

REMOTE TERMINAL TYPE

Description	Type of remote H.323 terminal.
Valid Value	TERMINAL GATEWAY MCU GATEKEEPER
Access	Read only.
Remarks	

DIRECTION

Description	The direction of call.
Valid Value	RX - incoming. TX - outgoing.
Access	Read only.
Remarks	

BITRATE

Description	Current call bit rate.
Valid Value	From 0 to max supported bit rate.
Access	Read only.
Remarks	

TX_VIDEO_CODEC

Description	The call's main outgoing video codec.
Valid Value	OFF H261 H263
Access	Read only.
Remarks	

RX VIDEO CODEC

Description	The call's main incoming video codec.
Valid Value	OFF H261 H263
Access	Read only.
Remarks	

TX VIDEO SEC CODEC

Description	The call's secondary outgoing video codec.
Valid Value	OFF H261 H263
Access	Read only.
Remarks	

RX VIDEO SEC CODEC

Description The call's secondary incoming video codec.

Valid Value **OFF**
 H261
 H263

Access Read only.

Remarks

TX AUDIO CODEC

Description The call's outgoing audio codec.

Valid Value **OFF**
G711A
G711U
G722
G723
G728
G729

Access Read only.

Remarks

RX AUDIO CODEC

Description The call's incoming audio codec.

Valid Value **OFF**
G711A
G711U
G722
G723
G728
G729

Access Read only.

Remarks

TX_VIDEO_BITRATE

Description	Actual main outgoing video bit rate.
Valid Value	From 0 to max supported bit rate.
Access	Read only.
Remarks	

RX_VIDEO_BITRATE

Description	Actual main incoming video bit rate.
Valid Value	From 0 to max supported bit rate.
Access	Read only.
Remarks	

TX_VIDEO_SEC_BITRATE

Description	Actual secondary outgoing video bit rate.
Valid Value	From 0 to max supported bit rate.
Access	Read only.
Remarks	

RX_VIDEO_SEC_BITRATE

Description	Actual secondary incoming video bit rate.
Valid Value	From 0 to max supported bit rate.
Access	Read only.
Remarks	

TX AUDIO BITRATE

Description	Actual outgoing audio bit rate.
Valid Value	From 0 to 64 Kb.
Access	Read only.
Remarks	

RX AUDIO BITRATE

Description	Actual incoming audio bit rate.
Valid Value	From 0 to 64 Kb
Access	Read only.
Remarks	

TX_VIDEO_HANDLE

Description	The main outgoing video codec handle.
Valid Value	Any value.
Access	Read only.
Remarks	

RX_VIDEO_HANDLE

Description	The main incoming video codec handle.
Valid Value	Any value.
Access	Read only.
Remarks	

TX_VIDEO_SEC_HANDLE

Description	The secondary outgoing video codec handle.
Valid Value	Any value.
Access	Read only.
Remarks	

RX_VIDEO_SEC_HANDLE

Description	The secondary incoming video codec handle.
Valid Value	Any value.
Access	Read only.
Remarks	

TX_AUDIO_HANDLE

Description	The outgoing audio codec handle.
Valid Value	Any value.
Access	Read only.
Remarks	

RX_AUDIO_HANDLE

Description	The incoming audio codec handle.
Valid Value	Any value.
Access	Read only.
Remarks	

TX_VIDEO_FRAMERATE

Description	The main outgoing video codec frame rate.
Valid Value	0 – 60 frames per second.
Access	Read only.
Remarks	

RX_VIDEO_FRAMERATE

Description	The main incoming video codec frame rate.
Valid Value	0 – 60 frames per second.
Access	Read only.
Remarks	

TX_VIDEO_SEC_FRAMERATE

Description	The secondary outgoing video codec frame rate.
Valid Value	0 – 60 frames per second.
Access	Read only.
Remarks	

RX_VIDEO_SEC_FRAMERATE

Description	The secondary incoming video codec frame rate.
Valid Value	0 – 60 frames per second.
Access	Read only.
Remarks	

DURATION

Description Call duration in seconds.

Valid Value value > 0.

Access Read only.

Remarks

REMOTE_NAME

Description	Name of remote endpoint.
Valid Value	Any value.
Access	Read only.
Remarks	

REMOTE_ADDRESS

Description	IP address of remote endpoint.
Valid Value	A valid IP address.
Access	Read only.
Remarks	

AUDIO TO VIDEO DELAY

Description	The delay between the incoming video and audio streams (in percent).
Valid Value	0 - 100 percent
Access	Read/Write.
Remarks	

JITTER BUFFER SIZE

Description	Returns the size of the used jitter buffer (in percent).
Valid Value	0 - 100 percent
Access	Read / Write.
Remarks	Relevant only when <code>AUTO_JITTER_ENABLE = 0</code>

TX VIDEO RESOLUTION

Description	The outgoing video resolution of the current codec.
Valid Value	QCIF - 176x144 CIF - 352x288 4CIF - 704x576 ICIF - 704x288 QSIF - 176x120 SIF - 352x240 ISIF - 704x240
Access	Read only.
Remarks	

RX VIDEO RESOLUTION

Description	The incoming video resolution of the current codec.
Valid Value	QCIF - 176x144 CIF - 352x288 4CIF - 704x576 ICIF - 704x288 QSIF - 176x120 SIF - 352x240 ISIF - 704x240
Access	Read only.
Remarks	

TX VIDEO SEC RESOLUTION

Description	The secondary outgoing video resolution of the current codec.
Valid Value	QCIF - 176x144 CIF - 352x288 4CIF - 704x576 ICIF - 704x288 QSIF - 176x120 SIF - 352x240 ISIF - 704x240
Access	Read only.
Remarks	

RX VIDEO SEC RESOLUTION

Description	The secondary incoming video resolution of the current codec.
Valid Value	QCIF - 176x144 CIF - 352x288 4CIF - 704x576 ICIF - 704x288 QSIF - 176x120 SIF - 352x240 ISIF - 704x240
Access	Read only.
Remarks	

CallSetBitrate

Description Set the video transmission rate for a specific call.

Format CallSetBitrate [*CallID=0*] <*Bitrate=384000*>

Parameters **CallID** – The ID of the call.
Bitrate – The new call bit per seconds.

Return **OK** on success.
ERROR on error together with an error code.

Example
CallSetBitrate 0 192000
CALLSETBITRATE OK

Remarks

CallSendIntra

Description	Send an intra frame to the remote party during a specific call.
Format	CallSendIntra [<i>CallID=0</i>]
Parameters	CallID – The ID of the call.
Return	OK on success. ERROR on error together with an error code.
Example	CallSendIntra 0 CALLSENDINTRA OK
Remarks	

CallRequestIntra

Description	Request an intra frame from the remote party during a specific call.
Format	CallRequestIntra [<i>CallID=0</i>]
Parameters	CallID – The ID of the call.
Return	OK on success. ERROR on error together with an error code.
Example	CallRequestIntra 0 CALLREQUESTINTRA OK
Remarks	

CallSendDTMF

Description	Send a DTMF tone to the remote party for a specific call.
Format	CallSendDTMF <CallID= 0 > <DTMF>
Parameters	CallID – The ID of the call. DTMF – The DTMF tone to send { 0..9 # * }
Return	OK on success. ERROR on error together with an error code.
Example	CallSendDTMF 0 3 CALLSENDDTMF OK
Remarks	

CallEvents

Description	Start or stop sending call event to the console.
Format	CallEvents [<i>IsEnable=1</i>]
Parameters	IsEnable – { 0 1 } 0 - Stop sending call event to the console. 1 - Start sending call event to the console.
Return	OK on success. ERROR on error together with an error code.
Example	CallEvents 1 CALLEVENTS OK
Remarks	Supported call events: EVENT_CALL_DIALTONE <CallID> EVENT_CALL_RINGBACK <CallID> EVENT_CALL_CONNECTED <CallID> EVENT_CALL_OFFERING <CallID> <RemoteAddress> <RemoteName> EVENT_CALL_DISCONNECTED <CallID> <DisconnectionReason> EVENT_CALL_IDLE <CallID> EVENT_CALL_CHANNEL_ON <CallID> <ChannelID> <Dir> <Type> EVENT_CALL_CHANNEL_OFF <CallID> < ChannelID > <Dir> <Type> EVENT_CALL_REMOTE_NOT_RESPONDING <CallID>

3.2 LAN Commands

GetParam

Description	Retrieve the value of a specific LAN configuration parameter.
Format	GetParam <LANParameter>
Parameters	LANParameter – The name of the LAN parameter.
Return	OK on success together with the LAN parameter value. ERROR on error together with an error code.
Example	GetParam IP_ADDRESS GETPARAM OK 10.0.11.25
Remarks	To retrieve all the LAN parameters, type GetParam . NULL string represents a null value. Supported LAN parameters: GATEKEEPER_LOGIN_STATE GATEKEEPER_ADDRESS STATION_NUMBER STATION_NAME SERVER_LOGIN_STATE SERVER_ADDRESS AUTO_ADDRESS MAC_ADDRESS IP_ADDRESS SUBNET_MASK DEFAULT_GATEWAY

**Remarks
(cont.)**

DNS_SERVER_ADDRESS
WINS_SERVER_ADDRESS
DOMAIN_NAME
NAT_ADDRESS
AUTO_ANSWER
LIPSYNC_ENABLE
AUTO_JITTER_ENABLE
ABA_ENABLE
MAX_BITRATE
MIN_BITRATE
DEF_BITRATE
USE_GATEKEEPER
USE_SERVER
MIN_UDP_PORT
MAX_UDP_PORT
MIN_TCP_PORT
MAX_TCP_PORT
AUDIO_QOS
VIDEO_QOS
RTCP_QOS
QOS_TYPE
AUTO_ACCEPT_MULTICAST_FLOOR
DHCP_STATE
WINS_LOGIN_STATE
STREAMING_ENABLE
STREAMING_ADDRESS
STREAMING_VIDEO_PORT

**Remarks
(cont.)**

STREAMING_AUDIO_PORT
STREAMING_BITRATE
STREAMING_TTL
STREAMING_TIMEOUT
STREAMING_ENABLE_ANNOUNCEMENT
ETHERNET_NIC_STATUS
ETHERNET_NIC_SETTINGS
MAX_ACTIVE_CALLS
ACTIVE_CALLS
FECC_ALLOWED
ANSWER_MODE_P2P
ANSWER_MODE_MULTIPPOINT
BAN_OUTGOING_CALLS
CRYPTO_ALGORITHM

Descriptions for the parameters listed above appear on the following pages.

GATEKEEPER LOGIN STATE

Description	Login state of the gatekeeper.
Valid Value	LOGGED_IN LOGIN_FAILED_DUPLICATE_ALIAS LOGIN_FAILED_GENERAL_ERROR LOGGED_OFF
Default Value	<i>LOGGED_OFF</i>
Access	Read only.
Set Effect	N/A.
Remarks	

GATEKEEPER ADDRESS

Description	IP address of the gatekeeper.
Valid Value	Should match IP addressing conventions (x.x.x.x where x= 0–255).
Default Value	NULL
Access	Read / Write.
Set effect	Reboot.
Remarks	

STATION_NUMBER

Description	E.164 alias of a H.323 station.
Valid Value	A string of digits.
Default Value	NULL
Access	Read / Write.
Set effect	Reboot.
Remarks	Dialing a STATION_NUMBER as a remote address is only relevant when endpoint is configured to work with a gatekeeper.

STATION_NAME

Description	Name of the endpoint.
Valid Value	String of characters, with a letter as the first character.
Default Value	NULL
Access	Read / Write.
Set effect	Reboot.
Remarks	Dialing a STATION_NAME as a remote address is only relevant when endpoint is configured to work with a gatekeeper.

SERVER LOGIN STATE

Description	The endpoint's login state in the MXM server.
Valid Value	LOGGED_IN LOGIN_FAILED_DUPLICATE_ALIAS LOGIN_FAILED_LICENSE_VIOLATION LOGIN_FAILED_UNSUPPORTED_VERSION LOGIN_FAILED_TIME_OUT LOGIN_FAILED_WAIT_FOR_GRANT LOGIN_FAILED_GENERAL_ERROR LOGGED_OFF
Default Value	<i>LOGGED_OFF</i>
Access	Read only.
Set effect	N/A.
Remarks	

SERVER ADDRESS

Description	The IP address of the MXM server.
Valid Value	Should match IP addressing conventions (x.x.x.x where x= 0–255).
Default Value	NULL
Access	Read / Write.
Set effect	Reboot.
Remarks	

AUTO ADDRESS

Description	Enable or disable automatic get of IP address from DHCP server.
Valid Value	{0 1} 0 – Disable. 1 – Enable.
Default Value	0
Access	Read / Write.
Set effect	Reboot.
Remarks	

MAC ADDRESS

Description	Returns the MAC address of the endpoint.
Valid Value	A string in the following format <i>XX:XX:XX:XX:XX</i> where <i>X</i> can be either a digit OR a letter in the following range A-F .
Default Value	N/A.
Access	Read only.
Set effect	N/A.
Remarks	

IP ADDRESS

Description	Local endpoint IP address.
Valid Value	Should match IP addressing conventions (x.x.x.x where x= 0–255).
Default Value	N/A.
Access	Read / Write.
Set effect	Reboot.
Remarks	

SUBNET MASK

Description	Subnet mask of the local endpoint.
Valid Value	Should match IP addressing conventions (x.x.x.x where x= 0–255).
Default Value	N/A.
Access	Read / Write.
Set effect	Reboot.
Remarks	

DEFAULT_GATEWAY

Description	The network's default gateway.
Valid Value	Should match IP addressing conventions (x.x.x.x where x= 0–255).
Default Value	N/A.
Access	Read / Write.
Set effect	Reboot.
Remarks	

DNS_SERVER_ADDRESS

Description	IP Address of network's DNS Server.
Valid Value	Should match IP addressing conventions (x.x.x.x where x= 0-255).
Default Value	NULL.
Access	Read / Write.
Set effect	Reboot.
Remarks	

WINS_SERVER_ADDRESS

Description	IP Address of network's WINS Server.
Valid Value	Should match IP addressing conventions (x.x.x.x where x= 0-255).
Default Value	NULL.
Access	Read / Write.
Set effect	Reboot.
Remarks	

DOMAIN_NAME

Description	The name of the domain in which the endpoint is located.
Valid Value	String of dot-separated name elements.
Default Value	NULL.
Access	Read / Write.
Set effect	Reboot.
Remarks	

NAT_ADDRESS

Description	The NAT IP address of the endpoint.
Valid Value	Should match IP addressing conventions (x.x.x.x where x= 0–255).
Default Value	NULL.
Access	Read / Write.
Set effect	Next Call.
Remarks	

AUTO ANSWER

Description	Enable/disable automatic answer of incoming call.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	1
Access	Read / Write.
Set effect	Next Call.
Remarks	

LIPSYNC_ENABLE

Description	Enable/disable lip synchronization mechanism.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	1
Access	Read / Write.
Set effect	Next Call.
Remarks	

AUTO JITTER ENABLE

Description	Enable/disable automatic buffering control, which allows manual or automatic control over delay and picture smoothness.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	1
Access	Read / Write.
Set effect	Next Call.
Remarks	

ABA_ENABLE

Description	Enable/disable Adaptive Bandwidth Adjustment mechanism for overcoming network congestion problems.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	1
Access	Read / Write.
Set effect	Next Call.
Remarks	

MAX_BITRATE

Description	Maximum bandwidth the system can allocate to initiate a call, according to its capabilities and/or license permissions.
Valid Value	0 – 2048000
Default Value	2048000
Access	Read / Write.
Set effect	Next Call.
Remarks	

MIN_BITRATE

Description	Minimum bandwidth the system can allocate to initiate a call, according to its capabilities and/or license permissions.
Valid Value	0 - 2048000
Default Value	64000
Access	Read / Write.
Set effect	Next Call.
Remarks	

DEF_BITRATE

Description	Default bandwidth the system can allocate to initiate a call, according to its capabilities and/or license permissions.
Valid Value	0 - 2048000
Default Value	768000
Access	Read / Write.
Set effect	Next Call.
Remarks	

USE_GATEKEEPER

Description	Enable or disable usage of gatekeeper management and services.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	0
Access	Read / Write.
Set effect	Next Reboot.
Remarks	

USE_SERVER

Description	Enable/disable usage of VCON Server management and services.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	0
Access	Read / Write.
Set effect	Next Reboot.
Remarks	

MIN_UDP_PORT

Description	Lowest UDP port allowed.
Valid Value	1024 - 65536
Default Value	5004
Access	Read / Write.
Set effect	Next Call.
Remarks	

MAX_UDP_PORT

Description	Highest UDP port allowed.
Valid Value	1024 - 65536
Default Value	6004
Access	Read / Write.
Set effect	Next Call.
Remarks	

MIN_TCP_PORT

Description	Lowest TCP port allowed.
Valid Value	1024 - 65536
Default Value	5004
Access	Read / Write.
Set effect	Next Call.
Remarks	

MAX TCP PORT

Description	Highest TCP port allowed.
Valid Value	1024 - 65536
Default Value	6004
Access	Read / Write.
Set effect	Next Call.
Remarks	

AUDIO_QOS

Description	The QOS value appended to each audio packet.
Valid Value	0 - 255
Default Value	160
Access	Read / Write.
Set effect	Next Call.
Remarks	

VIDEO_QOS

Description	The QOS value appended to each video packet.
Valid Value	0 - 255
Default Value	128
Access	Read / Write.
Set effect	Next Call.
Remarks	

RTCP_QOS

Description	The QOS value appended to each RTCP packet.
Valid Value	0 - 255
Default Value	192
Access	Read / Write.
Set effect	Next Call.
Remarks	

QOS TYPE

Description	Type of QOS which is currently used during the endpoint's calls.
Valid Value	NONE IPPRECEDENCE DIFFSERV
Default Value	<i>IPPRECEDENCE</i>
Access	Read / Write.
Set effect	Next Call.
Remarks	

AUTO ACCEPT MULTICAST FLOOR

Description	Automatically accept a multicast floor grant from the chair.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	1
Access	Read / Write.
Set effect	Next Call.
Remarks	

DHCP STATE

Description	The endpoint received or did not receive IP address from the DHCP server.
Valid Value	NOT_IN_USE RECEIVED_ADDRESS FAILED_RECEIVING_ADDRESS
Default Value	NOT_IN_USE
Access	Read only.
Set effect	N/A.
Remarks	

WINS LOGIN STATE

Description	The endpoint's login state in the WINS server.
Valid Value	LOGGED_OFF LOGIN_IN LOGIN_FAILED_NAME_FORMAT_ERROR LOGIN_FAILED_GENERAL_ERROR LOGIN_FAILED_NAME_ALREADY_USED LOGIN_FAILED_SERVER_NOT_RESPONDING
Default Value	<i>LOGGED_OFF</i>
Access	Read only.
Set effect	N/A.
Remarks	

STREAMING_ENABLE

Description	The streaming option is enabled or disabled.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	1
Access	Read \ Write.
Set effect	Next Call.
Remarks	

STREAMING ADDRESS

Description	The destination multicast address for streaming video and audio.
Valid Value	Should match IP addressing conventions (<i>x.x.x.x</i> where <i>x=0–255</i>).
Default Value	239.XXX.YYY.ZZZ
Access	Read \ Write.
Set effect	Next Call.
Remarks	

STREAMING VIDEO PORT

Description	The destination port for the streaming video.
Valid Value	1024 – 65536
Default Value	36100
Access	Read \ Write.
Set effect	Next Call.
Remarks	

STREAMING AUDIO PORT

Description	The destination port for the streaming audio.
Valid Value	1024 – 65536
Default Value	18100
Access	Read \ Write.
Set effect	Next Call.
Remarks	

STREAMING_BITRATE

Description	The bit rate of the streaming session.
Valid Value	0 - 2048000
Default Value	384000
Access	Read \ Write.
Set effect	Next Call.
Remarks	

STREAMING_TTL

Description	The time-to-live value appended to each multicast streaming packet.
Valid Value	0 - 255
Default Value	1
Access	Read \ Write.
Set effect	Next Call.d
Remarks	

STREAMING_TIMEOUT

Description	Unicast streaming stops automatically if the remote viewer has not requested resume-streaming within this time value.
Valid Value	<i>value</i> > 0 (in milliseconds)
Default Value	1800000
Access	Read \ Write.
Set effect	Next Call.
Remarks	Relevant to unicast streaming sessions only.

STREAMING ENABLE ANNOUNCEMENT

Description	Send SDP announcement for a multicast streaming session.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	1
Access	Read \ Write.
Set effect	Next Call.
Remarks	

ETHERNET NIC STATUS

Description	The resulting link speed
Valid Value	10HD 10FD 100HD 100FD
Default Value	N/A
Access	Read only.
Set effect	
Remarks	HD = Half Duplex FD = Full Duplex

ETHERNET NIC SETTINGS

Description	Get the current speed at which the HD is configured to work.
Valid Value	10HD 10FD 100HD 100FD AUTO - Auto negotiation
Default Value	N/A
Access	Read/write.
Set effect	Reboot
Remarks	HD = Half Duplex FD = Full Duplex

MAX_ACTIVE_CALLS

Description	Get the maximum number of concurrent calls supported.
Valid Value	<i>value</i> > 0
Default Value	3
Access	Read only.
Set effect	N/A.
Remarks	

ACTIVE CALLS

Description	Get the set of active CallIDs.
Valid Value	See Remarks section.
Default Value	NULL
Access	Read only.
Set effect	N/A.
Remarks	<p>For example:</p> <p>If the endpoint is running two active calls with IDs 1 & 2, the result is:</p> <pre>SetParam ACTIVE_CALLS SETPARAM OK 1 2</pre>

FECC_ALLOWED

Description	Enable remote party to control the local camera.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	1
Access	Read \ Write.
Set effect	Next Call.
Remarks	SetParam FECC_ALLOWED 1 SETPARAM OK

ANSWER MODE P2P

Description	Set the point-to-point answer mode
Valid Value	MANUAL_ANSWER AUTO_ANSWER AUTO_ANSWER_MUTE_MIC AUTO_REJECT
Default Value	N/A
Access	Read \ Write.
Set effect	Next Call.
Remarks	

ANSWER MODE MULTIPOINT

Description	Set the multipoint answer mode
Valid Value	MANUAL_ANSWER AUTO_ANSWER AUTO_REJECT
Default Value	N/A
Access	Read \ Write.
Set effect	Next Call.
Remarks	

BAN OUTGOING CALLS

Description	Prevent the user from making outgoing calls.
Valid Value	{0 1} 0 - Disable 1 - Enable
Default Value	N/A
Access	Read/write.
Set effect	Next call.
Remarks	

CRYPTO ALGORITHM

Description	Set the crypto algorithm for the next call.
Valid Value	{ NONE AUTO AES } NONE - HD will operate in clear mode. AUTO - HD will try to establish an encrypted call using AES. If unsuccessful, the call will not be encrypted. AES - HD will try to establish an encrypted call using AES. If unsuccessful, the call connection will disconnect.
Default Value	N/A
Access	Read/write.
Set effect	Next call.
Remarks	

SetParam

Description	Edit the value of a LAN configuration parameter.
Format	GetParam <LANParameter> <Value>
Parameters	LANParameter – The name of the LAN parameter to edit. Value – The new value
Return	OK on success together with the expected result and an indication of when the effect occurs. ERROR on error together with an error code.
Example	<pre>SetParam IP_ADDRESS 10.0.11.25 SETPARAM OK EFFECT_REBOOT</pre>
Remarks	To retrieve all the LAN parameters, type SetParam . For a list of LAN parameters, see “GetParam” on page 60 . NULL string represents a null value. Supported Effects: EFFECT_NONE — No change was made. EFFECT_IMMEDIATE — Effect occurs immediately. EFFECT_NEXT_CALL — Effect occurs during the next call. EFFECT_REBOOT — Effect occurs after the computer restarts.

ServerEvents

Description	Start or stop sending server event to the console.
Format	ServerEvents [<i>IsEnable=1</i>]
Parameters	IsEnable – {0 1} 0 - Stop sending server events to the console. 1 - Start sending server events to the console.
Return	OK on success. ERROR on error together with an error code.
Example	ServerEvents 1 SERVEREVENTS OK
Remarks	Supported server events: EVENT_SERVER_DATA_CHANGED <Action> EVENT_SERVER_COMMAND <Command> EVENT_SERVER_STATE_CHANGED <NewState>

GatekeeperEvents

Description	Start or stop sending gatekeeper event to the console.
Format	GatekeeperEvents [<i>IsEnable=1</i>]
Parameters	IsEnable – {0 1} 0 - Stop sending gatekeeper events to the console. 1 - Start sending gatekeeper events to the console.
Return	OK on success. ERROR on error together with an error code.
Example	GatekeeperEvents 1 GATEKEEPEREVENTS OK
Remarks	Gatekeeper supported events: EVENT_GATEKEEPER_STATE_CHANGED <NewState>

3.3 Streaming Commands

StreamingStart

Description	Start a new streaming session.
Format	StreamingStart [<i>RemoteAddress</i> =" "]
Parameters	RemoteAddress – The IP address to where to send the media.
Return	OK on success together with a SessionID. ERROR on error together with an error code.
Example	<pre>StreamingStart STREAMINGSTART OK 0</pre>
Remarks	<ol style="list-style-type: none">1. When the RemoteAddress is NULL a multicast session will be created.2. If the value is a valid IP a unicast streaming session will be created.3. There may be only one Multicast session. For the maximum number of permitted unicast sessions, see the applicable HD device data sheet.

StreamingStop

Description	End an active streaming session.
Format	StreamingStop <SessionID>
Parameters	SessionID – The ID of the session to stop.
Return	OK on success. ERROR on error together with an error code.
Example	StreamingStop 1 STREAMINGSTOP OK
Remarks	

StreamingResume

Description	Resume an active streaming session.
Format	StreamingResume <SessionID>
Parameters	SessionID – The ID of the streaming session.
Return	OK on success together with a new resume timeout. ERROR on error together with an error code.
Example	<pre>StreamingResume 1 STREAMINGRESUME OK 180000</pre>
Remarks	Unicast streaming session has a watchdog mechanism. Users must call this function periodically at intervals smaller than the interval returned by this command; otherwise, the unicast session shall be terminated by the HD.

StreamingEvents

Description	Start or stop sending streaming event to the console.
Format	StreamingEvents [<i>IsEnable=1</i>]
Parameters	IsEnable – {0 1} 0 - Stop sending streaming event to the console. 1 - Start sending streaming event to the console.
Return	OK on success. ERROR on error together with an error code.
Example	<pre>StreamingEvents 1 STREAMINGEVENTS OK</pre>
Remarks	Streaming supported events: EVENT_STREAMING_STARTED <SessionID> <IsFirstSession> EVENT_STREAMING_STOPPED <SessionID> <IsLastSession>

3.4 *Interactive Multicast Commands*

MulticastSendPassword

Description	Send the multicast password to the multicast chair.
Format	MulticastSendPassword [<i>CallID=0</i>] < <i>Password</i> >
Parameters	CallID – The ID of the call. Password – The session’s password.
Return	OK on success. ERROR on error together with an error code.
Example	MulticastSendPassword 0 1234 MULTICASTSENDPASSWORD OK
Remarks	This function is the required response to the EVENT_MULTICAST_PASSWORD_REQUESTED event received from the multicast chair.

MulticastSendText

Description	Send a text message to the multicast chair.
Format	MulticastSendText [<i>CallID=0</i>] < <i>Text</i> >
Parameters	CallID – The ID of the call. Text – The text message.
Return	OK on success. ERROR on error together with an error code.
Example	<pre>MulticastSendText 0 "Hello Chair" MULTICASTSENDEXT OK</pre>
Remarks	

MulticastRequestFloor

Description	Request the multicast floor from the chair.
Format	MulticastRequestFloor [<i>CallID=0</i>]
Parameters	CallID – The ID of the call.
Return	OK on success. ERROR on error together with an error code.
Example	MulticastRequestFloor 0 MULTICASTREQUESTFLOOR OK
Remarks	

MulticastRejectFloor

Description	Reject a floor grant from the chair.
Format	MulticastRejectFloor [<i>CallID=0</i>]
Parameters	CallID – The ID of the call.
Return	OK on success. ERROR on error together with an error code.
Example	MulticastRejectFloor 0 MULTICASTREJECTFLOOR OK
Remarks	

MulticastAcceptFloor

Description	Accept a floor grant from the chair.
Format	MulticastAcceptFloor [<i>CallID=0</i>]
Parameters	CallID – The ID of the call.
Return	OK on success. ERROR on error together with an error code.
Example	MulticastAcceptFloor 0 MULTICASTACCEPTFLOOR OK
Remarks	

MulticastEvents

Description	Start or stop sending multicast event to the console.
Format	MulticastEvents [<i>IsEnable=1</i>]
Parameters	IsEnable – {0 1} 0 - Stop sending multicast event to the console. 1 - Start sending multicast event to the console.
Return	OK on success. ERROR on error together with an error code.
Example	<pre>MulticastEvents 1 MULTICASTEVENTS OK</pre>
Remarks	Supported multicast events: EVENT_MULTICAST_CONNECTED <CallID> EVENT_MULTICAST_FLOOR_OFFERING <CallID> EVENT_MULTICAST_FLOOR_REQUEST_REJECTED <CallID> EVENT_MULTICAST_FLOOR_CHANGED <CallID> <Type> <Name> EVENT_MULTICAST_FLOOR_GRANTED <CallID> EVENT_MULTICAST_PASSWORD_REJECTED <CallID> EVENT_MULTICAST_PASSWORD_REQUESTED <CallID>

3.5 Camera Commands

CameraGetSelected

Description	Get the selected local or remote camera.
Format	CameraGetSelected <Location={l r}>
Parameters	Location – {l r} l - local camera. r - remote camera.
Return	OK on success together with the selected camera number. ERROR on error together with an error code.
Example	CameraGetSelected l CAMERAGETSELECTED OK 1
Remarks	

CameraSetSelected

Description	Switch the selected camera to either the local or remote camera.
Format	CameraSetSelected <Location={l r}> <CameraNumber={1..4}>
Parameters	Location – {l r} l - local camera. r - remote camera. CameraNumber 1..4 - The selected camera number
Return	OK on success. ERROR on error together with an error code.
Example	CameraSetSelected l 3 CAMERASETSELECTED OK
Remarks	The “Main” camera’s number is 1.

CameraPresetSet

Description	Set the selected local or remote camera's preset position.
Format	CameraPresetSet <Location={l r}> <PresetNumber={0..9}>
Parameters	Location – {l r} l - local camera. r - remote camera. PresetNumber – {0..9} 0..9 - The preset position number
Return	OK on success. ERROR on error together with an error code.
Example	CameraPresetSet l 9 CAMERAPRESETSET OK
Remarks	

CameraPresetRecall

Description	Move the selected local or remote camera to one of the preset positions.
Format	CameraPresetRecall <Location={l r}> <PresetNumber={0..9}>
Parameters	Location – {l r} l - local camera. r - remote camera. PresetNumber – {0..9} 0..9 - The preset position number
Return	OK on success. ERROR on error together with an error code.
Example	CameraPresetRecall l 3 CAMERAPRESETRECALL OK
Remarks	

CameraMove

Description	Move the camera's position.
Format	CameraMove <Location={l r}> <Dir={u d r l i o s}>
Parameters	Location – {l r} l - local camera. r - remote camera. Dir – The direction in which to move the camera u d r l i o s u - up d - down r - right l left i - in o - out s - stop
Return	OK on success. ERROR on error together with an error code.
Example	CameraMove l u CAMERAMOVE OK
Remarks	

3.6 Video Commands

MuteVideo

Description	Mute the video sent to the remote side, which sees a predefined bitmap instead of your local video.
Format	MuteVideo [<i>Enable</i> ={0 1}]
Parameters	Enable - start/stop sending video to the remote side. 1 = Mute the local video. 0 = Stop muting and resume sending local video to the remote side.
Return	OK ERROR on error together with an error code.
Example	MuteVideo 1 MUTEVIDEO OK

DisplaySet

Description	Control the display layout of the HD
Format	DisplaySet <Dest={ window1 pip1 }> <Source={ local1 remote1 openpicture data }>
Parameters	Dest - defines the area of the video display to change. window1 = main display area of the TV pip1 = default PIP Source - defines what to display in the “Dest” area. local1 : your local video remote1 : remote side’s video openpicture : HD startup bitmap data : display of the remote PC desktop being broadcast through the HD Data Utility.
Return	OK ERROR on error together with an error code.
Example	DisplaySet window1 local1 OK

3.7 Audio Commands

AudioSetMicInput

Description	Select the audio input device.
Format	AudioSetMicInput <Mic= TABLETOP LINELEVEL >
Parameters	TABLETOP - tabletop microphone LINELEVEL - auxiliary audio source, such as DVD, VCR.
Return	OK on success. ERROR on error together with an error code.
Example	<pre>AudioSetMicInput TABLETOP AUDIOSETMICINPUT OK</pre>
Remarks	

AudioGetMicInput

Description	Get the selected audio input device.
Format	AudioGetMicInput
Parameters	
Return	OK on success together with the selected input device. ERROR on error together with an error code.
Example	AudioGetMicInput AUDIOGETMICINPUT OK TABLETOP
Remarks	

AudioSetMicGainLevel

Description Increase amplification from the audio input device.

Format `AudioSetMicGainLevel <Level>`

Parameters **Level** - the amplification level {**0-7**}

Return **OK** on success.
ERROR on error together with an error code.

Example `AudioSetMicGainLevel 3`
`AUDIOSETMICGAINLEVEL OK`

Remarks

AudioGetMicGainLevel

Description Get the amplification level of the selected audio input device.

Format AudioGetMicGainLevel

Parameters

Return **OK** on success with the amplification level
ERROR on error together with an error code.

Example AudioGetMicGainLevel
AUDIOGETMICGAINLEVEL OK 3

Remarks

AudioFilePlay

Description	Play a sound file.
Format	AudioFilePlay <FileName> <Location>
Parameters	<p>FileName - the name of the audio file to play. The audio file must reside in the [TONES] directory in the compact flash. The file should contain raw 16-bit PCM audio samples.</p> <p>Location = LOCAL REMOTE LOCAL_AND_REMOTE The end point that will play the audio. The audio file may be played either to the local side, sent to the remote side, or to all sides.</p>
Return	<p>OK on success.</p> <p>ERROR on error together with an error code.</p>
Example	<pre>AudioFilePlay dial.pcm LOCAL AUDIOFILEPLAY OK</pre>

AudioFileStop

Description	Stop playing an audio file.
Format	AudioFileStop
Parameters	
Return	OK on success. ERROR on error together with an error code.
Example	AudioFileStop AUDIOFILESTOP OK
Remarks	

MuteMic

Description	Mute or unmute the microphone.
Format	MuteMic [<i>Enable</i> = 1]
Parameters	Enable: { 0 1 } 0 – Unmute 1 – Mute
Return	OK on success. ERROR on error together with an error code.
Example	MuteMic 1 MUTEMIC OK
Remarks	

MuteSpeaker

Description	Mute or unmute the speaker.
Format	MuteSpeaker [<i>Enable=1</i>]
Parameters	Enable – {0 1} 0 – Unmute 1 – Mute
Return	OK on success. ERROR on error together with an error code.
Example	MuteSpeaker 1 MUTESPEAKER OK
Remarks	

VolumeGet

Description	Get the current volume level.
Format	VolumeGet
Parameters	N/A
Return	OK on success together with the current volume level. ERROR on error together with an error code.
Example	VolumeGet VOLUMEGET OK 60
Remarks	

VolumeSet

Description	Change the current volume level.
Format	VolumeSet <Level>
Parameters	Level - The volume level { 0-99 }. 0 - Lowest. 99 - Highest.
Return	OK on success. ERROR on error together with an error code.
Example	VolumeSet 50 VOLUMESET OK
Remarks	

3.8 Phonebook Commands

PhonebookGetEntry

Description Get the phonebook entry that matches the specified number.

Format PhonebookGetEntry <EntryNum>

Parameters **EntryNum** - The number of the entry to get.

Return **OK** on success together with the entry details.
ERROR on error together with an error code.

Example

```
PhonebookGetEntry 1
PHONEBOOKGETENTRY OK JohnY 10.0.11.24
768000
```

Remarks

PhonebookGetNumOfEntries

Description	Get the number of entries in the phonebook.
Format	PhonebookGetNumOfEntries
Parameters	N/A
Return	OK on success together with the number of entries in the phonebook. ERROR on error together with an error code.
Example	PhonebookGetNumOfEntries PHONEBOOKGETNUMOFENTRIES OK 100
Remarks	

PhonebookAddEntry

Description	Add a new entry to the phonebook.
Format	PhonebookAddEntry <Name> <Address> [Bitrate= 768000]
Parameters	Name - The name of the entry. Address - The remote party address (IP , DNS name , E164 , H323-ID). Bitrate - The bit rate of the call.
Return	OK on success. ERROR on error together with an error code.
Example	PhonebookAddEntry PHONEBOOKADDEENTRY OK
Remarks	

PhonebookDeleteEntry

Description	Delete the specified entry.
Format	PhonebookDeleteEntry < <i>Name</i> >
Parameters	Name - The name of the entry.
Return	OK on success. ERROR on error together with an error code.
Example	PhonebookDeleteEntry JohnY PHONEBOOKDELETEENTRY OK
Remarks	

PhonebookShow

Description	Show the content of the phonebook.
Format	PhonebookShow
Parameters	N/A
Return	OK on success. ERROR on error together with an error code.

Example	PhonebookShow												
	<table><thead><tr><th>##</th><th>Name</th><th>Address</th><th>Bitrate</th></tr></thead><tbody><tr><td>0.</td><td>JohnY</td><td>10.0.11.24</td><td>768000</td></tr><tr><td>1.</td><td>DavidS</td><td>10.0.11.25</td><td>384000</td></tr></tbody></table>	##	Name	Address	Bitrate	0.	JohnY	10.0.11.24	768000	1.	DavidS	10.0.11.25	384000
##	Name	Address	Bitrate										
0.	JohnY	10.0.11.24	768000										
1.	DavidS	10.0.11.25	384000										

Remarks

PhonebookDial

Description	Dial to the address of the specified entry.
Format	PhonebookDial <EntryName>
Parameters	EntryName - The name of the specific entry.
Return	OK on success together with the CallID. ERROR on error together with an error code.
Example	PhonebookDial JohnY PHONEBOOKDIAL OK 0
Remarks	

3.9 System Commands

IniRead

Description Read a configuration entry value from an INI file. The INI is in the HD non-volatile memory.

Format IniRead <FileName> <Section> <Entry>

Parameters **FileName** – The INI file name.

Section – The section name.

Entry – The entry name.

Return **OK** on success together with the entry value.

ERROR on error together with an error code.

Example IniRead H323.ini Configuration AutoIP
INIREAD OK 1

Remarks

IniWrite

Description	Write a configuration entry value to an INI file.
Format	IniWrite <FileName> <Section> <Entry> <Value>
Parameters	FileName – The INI file name. Section – The section name. Entry – The entry name. Value – The new value to be written.
Return	OK on success. ERROR on error together with an error code.
Example	<pre>IniWrite H323.ini Configuration AutoIP 0 INIWRITE OK</pre>
Remarks	

Reboot

Description	Reboot the system.
Format	Reboot
Parameters	None.
Return	OK on success. ERROR on error together with an error code.
Example	Reboot REBOOT OK
Remarks	

KeyPressed

Description	Simulate the pressing of a remote control key.
Format	<code>KeyPressed <KeyName></code>
Parameters	KeyName - The name of the key that was pressed.
Return	OK on success. ERROR on error together with an error code.
Example	<pre>KeyPressed Dial KEYPRESSED OK</pre>
Remarks	To get all the supported key names, type KeyPressed . The supported key names are: Zero One Two Three Four Five Six Seven Eight Nine Asterisk Pound Clear Display PIP Dial

**Remarks
(cont.)**

Hangup
Menu
Multipoint
Status
Help
OK
Cancel
Select
Pound
Up
Left
Down
Right
ZoomIn
ZoomOut
MuteMic
MuteSpeaker
VolumePlus
VolumeMinus
Red
Green
Blue
Yellow

ConsoleGetParam

Description	Retrieve a console parameter.
Format	ConsoleGetParam < <i>ConsoleParam</i> >
Parameters	ConsoleParam - The specific console parameter.
Return	OK on success together with the parameter value. ERROR on error together with an error code.
Example	ConsoleGetParam BAUD_RATE CONSOLEGETPARAM OK 9600
Remarks	To get a list of supported commands, type ConsoleGetParam . Supported call parameters: COM_NUMBER BAUD_RATE

ConsoleSetParam

Description	Edit a console parameter.
Format	ConsoleSetParam <ConsoleParam> <Value>
Parameters	ConsoleParam - The specific console parameter. Value - The value that will replace the current value.
Return	OK on success together with the parameter value. ERROR on error together with an error code.
Example	<pre>ConsoleSetParam BAUD_RATE 115200 CONSOLESETPARAM OK</pre>
Remarks	To get a list of supported commands, type ConsoleSetParam . Supported call parameters: COM_NUMBER BAUD_RATE

DiskInfo

Description Print the amount of free space on the HD compact flash non-volatile storage device.

Format DiskInfo

Return **OK** with the amount of free disk space.
ERROR on error together with an error code.

Example DiskInfo
There are 25690112 bytes free in device
cf0:

Ping

Description Emulate Windows's "ping" command, which checks if the remote device is "alive."

Format Ping <*IP Address*>

Parameters **IP Address** - IP address of the remote device that we want to test.

Return **OK**

ERROR on error together with an error code.

Example Ping 172.20.1.1

Replay from 172.20.1.1 ,bytes=64 ,times=0

ShowDbgMsg

Description	Start/stop sending the debug log from the HD to the HDK device (for example, Hyper Terminal).
Format	ShowDbgMsg [<i>Enable=1</i>]
Parameters	Enable - {0 1} 1 = Start getting log messages from the HD device. 0 = Stop getting log messages.
Return	OK on success. ERROR on error together with an error code.
Example	ShowDbgMsg 1 Start Sending log messages to the ZConsole application.

4 *EVENTS*

This chapter contains description, syntaxes, parameters, and examples of the HDK events.

4.1 *Call Events*

EVENT_CALL_DIALTONE

Description	A dialtone state followed dialing.
Format	EVENT_CALL_DIALTONE <CallID>
Parameters	CallID – The ID of the call.
Example	EVENT_CALL_DIALTONE 0
Remarks	<ol style="list-style-type: none">1. Relevant for outgoing calls only.2. This is the first event for outgoing calls.

EVENT_CALL_RINGBACK

Description	An “Alerting” or “Proceeding” message was received from the remote side.
Format	EVENT_CALL_RINGBACK <CallID>
Parameters	CallID – The ID of the call.
Example	EVENT_CALL_RINGBACK 0
Remarks	Relevant for outgoing calls only.

EVENT_CALL_CONNECTED

Description	A call connects successfully.
Format	EVENT_CALL_CONNECTED <CallID>
Parameters	CallID – The ID of the call.
Example	EVENT_CALL_CONNECTED 0
Remarks	Relevant for both outgoing and incoming calls.

EVENT_CALL_OFFERING

Description	An incoming call indication was received.
Format	EVENT_CALL_OFFERING <CallID> <RemoteAddress> <RemoteName>
Parameters	CallID – The ID of the call RemoteAddress – The remote party IP address RemoteName – The remote party name
Example	EVENT_CALL_OFFERING 0 10.0.11.25 DemoRoom
Remarks	<ol style="list-style-type: none">1. Relevant for incoming calls.2. When working in manual answer the user will need to call either CallAnswer or CallReject commands.

EVENT_CALL_DISCONNECTED

Description	A call was disconnected.
Format	EVENT_CALL_DISCONNECTED <CallID> <DisconnectionReason>
Parameters	CallID – The ID of the call DisconnectionReason – The disconnection reason
Example	EVENT_CALL_DISCONNECTED 0 REMOTE
Remarks	<ol style="list-style-type: none">1. Relevant for incoming and outgoing calls.2. Disconnection reason<ul style="list-style-type: none">— LOCAL – Call disconnected by local party— REMOTE – Call disconnected by remote party— BUSY – Remote party is busy— REJECT - Remote party rejected the call— UNREACHABLE – Remote party is unreachable— NO_ANSWER – Remote party hasn't answered the calls— UNKNOWN – Call disconnected due to unknown reason.

EVENT_CALL_IDLE

Description	Fired after a call is disconnected. The disconnected call's resources are no longer valid.
Format	EVENT_CALL_IDLE <CallID>
Parameters	CallID – The ID of the call
Example	EVENT_CALL_IDLE 0
Remarks	Relevant for incoming and outgoing calls.

EVENT_CALL_CHANNEL_ON

Description	A new channel has been established on a call.
Format	EVENT_CALL_CHANNEL_ON <CallID> <ChannelID> <Dir> <Type>
Parameters	CallID – The ID of the call ChannelID – The ID of the Channel Dir – The Channel Direction { RX TX } Type – The Channel media type { AUDIO VIDEO DATA }
Example	EVENT_CALL_CHANNEL_ON 0 0 TX AUDIO EVENT_CALL_CHANNEL_ON 0 1 TX VIDEO EVENT_CALL_CHANNEL_ON 0 2 TX DATA EVENT_CALL_CHANNEL_ON 0 3 RX AUDIO EVENT_CALL_CHANNEL_ON 0 4 RX VIDEO EVENT_CALL_CHANNEL_ON 0 5 RX DATA
Remarks	Relevant for incoming and outgoing calls.

EVENT_CALL_CHANNEL_OFF

Description	An active channel was disconnected on a call.
Format	EVENT_CALL_CHANNEL_OFF <CallID> <ChannelID> <Dir> <Type>
Parameters	CallID – The ID of the call ChannelID – The ID of the Channel Dir – The Channel Direction (RX TX) Type – The Channel media type (AUDIO VIDEO DATA)
Example	EVENT_CALL_CHANNEL_OFF 0 0 TX AUDIO EVENT_CALL_CHANNEL_OFF 0 1 TX VIDEO EVENT_CALL_CHANNEL_OFF 0 2 TX DATA EVENT_CALL_CHANNEL_OFF 0 3 RX AUDIO EVENT_CALL_CHANNEL_OFF 0 4 RX VIDEO EVENT_CALL_CHANNEL_OFF 0 5 RX DATA
Remarks	Relevant for incoming and outgoing calls.

EVENT_CALL_REMOTE_NOT_RESPONDING

Description	During an active call, no signal is received from the remote side.
Format	EVENT_CALL_REMOTE_NOT_RESPONDING <CallID>
Parameters	CallID – The ID of the call
Example	EVENT_CALL_REMOTE_NOT_RESPONDING 0
Remarks	<ol style="list-style-type: none">1. Relevant for outgoing and incoming calls only.2. The application may disconnect the call or display a message to the user.

4.2 Server Events

EVENT_SERVER_DATA_CHANGED

Description Configuration data was changed by the MXM server.

Format EVENT_SERVER_DATA_CHANGED <Action>

Parameters **Action** –The current action the application should perform.

REBOOT the system.

REFRESH the application dialog.

Example

```
EVENT_SERVER_DATA_CHANGED REBOOT  
EVENT_SERVER_DATA_CHANGED REFRESH
```

Remarks

EVENT_SERVER_COMMAND

Description A server command was received and must be addressed by the application.

Format EVENT_SERVER_COMMAND <*Command*>

Parameters **Command** – The command the application should perform.

REBOOT the system.

Example EVENT_SERVER_COMMAND REBOOT

Remarks

EVENT_SERVER_STATE_CHANGED

Description	The endpoint's login state with the MXM server has changed.
Format	EVENT_SERVER_STATE_CHANGED <NewState>
Parameters	NewState – The current state with the MXM server.
Example	EVENT_SERVER_STATE_CHANGED LOGGED_IN
Remarks	<p>Possible states with the MXM:</p> <ol style="list-style-type: none">1. LOGGED_IN: Endpoint is logged in the the MXM server.2. LOGIN_FAILED_DUPLICATE_ALIAS: Another end point with the same H.323-ID E.164 is already logged in to the MXM.3. LOGIN_FAILED_UNSUPPORTED_VERSION: The MXM version doesn't support the endpoint.4. LOGIN_FAILED_LICENSE_VIOLATION: Failed to login due to license violation.5. LOGIN_FAILED_TIME_OUT: The login timeout has expired.6. LOGIN_FAILED_WAIT_FOR_GRANT: The endpoint is waiting for administrator to grant login.7. LOGIN_FAILED_GENERAL_ERROR: Failed to login due to unspecified reason.8. LOGGED_OFF: The endpoint isn't configured to work with MXM server.

4.3 Gatekeeper Events

EVENT_GATEKEEPER_STATE_CHANGED

Description	The endpoint's login state with the gatekeeper has changed.
Format	EVENT_GATEKEEPER_STATE_CHANGED <NewState>
Parameters	NewState – The current state with the gatekeeper
Example	EVENT_GATEKEEPER_STATE_CHANGED LOGGED_IN
Remarks	Possible states with the gatekeeper: <ol style="list-style-type: none">1. LOGGED_IN: Endpoint is logged in to the Gatekeeper.2. LOGIN_FAILED_DUPLICATE_ALIAS: Another end point with the same H.323-ID E.164 is already logged in to the gatekeeper.3. LOGIN_FAILED_GENERAL_ERROR: Failed to login due to an unspecified reason.4. LOGGED_OFF: The endpoint isn't configured to work with this gatekeeper.

4.4 Multicast Events

VCON developed a type of H.323 session called Interactive Multicast, in which one end point (*Chair*) transmits the same video and audio at a single time to all the other end points (*Participants*) in the session.

The Chair manages one “floor” token. An end point must have the token in order to speak. End points may request the token and wait for the Chair to accept its request. In addition, the Chair may give the token to any end point without a specific end point request.

EVENT_MULTICAST_CONNECTED

Description	Fired after a multicast session is connected.
Format	EVENT_MULTICAST_CONNECTED <CallID>
Parameters	CallID – The ID of the call.
Example	EVENT_MULTICAST_CONNECTED 0
Remarks	

EVENT_MULTICAST_FLOOR_OFFERING

Description	Fired when the multicast chair offers the floor to this endpoint. Only the endpoint who has the multicast floor can broadcast video and audio to all the other endpoints in the session.
Format	EVENT_MULTICAST_FLOOR_OFFERING <CallID>
Parameters	CallID – The ID of the call.
Example	EVENT_MULTICAST_FLOOR_OFFERING 0
Remarks	Application may call MulticastAcceptFloor to accept the floor or MulticastRejectFloor to reject it.

EVENT_MULTICAST_FLOOR_REQUEST_REJECTED

Description	Fired when the multicast chair rejected this endpoint's floor request.
Format	EVENT_MULTICAST_FLOOR_REQUEST_REJECTED <CallID>
Parameters	CallID – The ID of the call
Example	EVENT_MULTICAST_FLOOR_REQUEST_REJECTED 0
Remarks	

EVENT_MULTICAST_FLOOR_CHANGED

Description	Fired when the floor is given to another endpoint.
Format	EVENT_MULTICAST_FLOOR_CHANGED <CallID> <Type> <Name>
Parameters	CallID – The ID of the call. Type – The type of floor that has been changed { VIDEO AUDIO }. Name – The name of the participant who now owns the floor.
Example	EVENT_MULTICAST_FLOOR_CHANGED 0 VIDEO DemoRoom
Remarks	

EVENT_MULTICAST_FLOOR_GRANTED

Description	Fired when the multicast chair has granted the floor to this endpoint.
Format	EVENT_MULTICAST_FLOOR_GRANTED <CallID> <Type>
Parameters	CallID – The ID of the call Type – The type of floor that has been granted (VIDEO AUDIO)
Example	EVENT_MULTICAST_FLOOR_GRANTED 0 VIDEO
Remarks	

EVENT_MULTICAST_PASSWORD_REJECTED

Description	Fired when this endpoint enters an incorrect password while attempting to join a multicast session.
Format	EVENT_MULTICAST_PASSWORD_REJECTED <CallID>
Parameters	CallID – The ID of the call.
Example	EVENT_MULTICAST_PASSWORD_REJECTED 0
Remarks	Fired as a result of a call to MulticastSendPassword .

EVENT_MULTICAST_PASSWORD_REQUESTED

Description	Fired when the endpoint must enter a password in order to join a multicast session.
Format	EVENT_MULTICAST_PASSWORD_REQUESTED <CallID>
Parameters	CallID – The ID of the call
Example	EVENT_MULTICAST_PASSWORD_REQUESTED 0
Remarks	Call MulticastSendPassword to provide the password

4.5 Streaming Events

EVENT_STREAMING_STARTED

Description	Fired when a streaming session starts.
Format	EVENT_STREAMING_STARTED <SessionID> <IsFirstSession>
Parameters	SessionID – The streaming session ID. IsFirstSession – {0 1} Set to 1 if this is the first active streaming session.
Example	EVENT_STREAMING_STARTED 0 1
Remarks	

EVENT_STREAMING_STOPPED

Description	Fired after the streaming session ends.
Format	EVENT_STREAMING_STOPPED < <i>SessionID</i> > < <i>IsLastSession</i> >
Parameters	SessionID – The ID of the streaming session. IsLastSession – { 0 1 } Set to 1 if this is the last active streaming session.
Example	EVENT_STREAMING_STOPPED 0 1
Remarks	

4.6 System Events

EVENT_SYSTEM_INIT_COMPLETED_SUCCESS

Description	Fired when the system initializes successfully.
Format	EVENT_SYSTEM_INIT_COMPLETED_SUCCESS
Parameters	None.
Example	EVENT_SYSTEM_INIT_COMPLETED_SUCCESS
Remarks	Calling some of the commands prior to receiving this event may result in an error.

EVENT_SYSTEM_INIT_COMPLETED_FAILURE

Description	Fired when the system fails to initialize.
Format	EVENT_SYSTEM_INIT_COMPLETED_FAILURE
Parameters	None.
Example	EVENT_SYSTEM_INIT_COMPLETED_FAILURE
Remarks	User may try to Reboot the system to recover.

EVENT_SYSTEM_DHCP_STATE_RECEIVED_ADDRESS

Description	Fired when the system received a valid IP address from the DHCP server.
Format	EVENT_SYSTEM_DHCP_STATE_RECEIVED_ADDRESS
Parameters	None.
Example	EVENT_SYSTEM_DHCP_STATE_RECEIVED_ADDRESS
Remarks	Relevant only when the endpoint is configured to work with a DHCP.

EVENT_DHCP_STATE_FAILED_RECEIVING_ADDRESS

Description	Fired when system failed to receive an IP address from the DHCP server.
Format	EVENT_DHCP_STATE_FAILED_RECEIVING_ADDRESS
Parameters	None.
Example	EVENT_DHCP_STATE_FAILED_RECEIVING_ADDRESS
Remarks	<ol style="list-style-type: none">1. Relevant only when the endpoint is configured to work with a DHCP.2. Notify the user, check that the HD is connected to the network, and the DHCP server is working properly.

4.7 FECC Events

CAMERA_REMOTE_MOVE

Description	Notification about remote camera's movement.
Format	CAMERA_REMOTE_MOVE <Direction>
Parameters	Direction – {STOP IN OUT UP DOWN LEFT RIGHT UP_LEFT UP_RIGHT DOWN_LEFT DOWN_RIGHT HOME}
Example	CAMERA_REMOTE_MOVE LEFT
Remarks	To enable this notification, set the CameraEvents entry to 1 in the <i>user.ini</i> file.

CAMERA_LOCAL_MOVE

Description	Notification about local camera's movement.
Format	CAMERA_LOCAL_MOVE <Direction> <Source>
Parameters	Direction – {STOP IN OUT UP DOWN LEFT RIGHT UP_LEFT UP_RIGHT DOWN_LEFT DOWN_RIGHT HOME}
Example	CAMERA_LOCAL_MOVE LEFT REMOTE
Remarks	To enable this notification, set the CameraEvents entry to 1 in the <i>user.ini</i> file.

5 THE USER.INI FILE

Description The *user.ini* file contains several configuration parameters that you may edit. The file is located in the HD system's */ver/ini* folder.

Parameters

- Configuration parameters:
 - CallEvents
 - MulticastEvents
 - StreamingEvents
 - GatekeeperEvents
 - ServerEvents
- GUI parameters:
 - HideAll
 - HideDlg

5.1 Configuration Parameters

CallEvents

Description	System sends all Call Events automatically. User does not need to manually call the CallEvents command.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	0

MulticastEvents

Description	System sends all Multicast Events automatically. User does not need to manually call the MulticastEvents command.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	0

StreamingEvents

Description	System sends all Streaming Events automatically. User does not need to manually call the StreamingEvents command.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	0

GatekeeperEvents

Description	System sends all Gatekeeper Events automatically. User does not need to manually call the GatekeeperEvents command.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	0

ServerEvents

Description	System sends all Server Events automatically. User does not need to manually call the ServerEvents command.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	0

5.2 GUI Parameters

HideAll

Description	Hides the HD device's user interface. Only video is displayed on the screen.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	0

HideDlg

Description	Hides the HD device's dialog boxes. Only video, messages and icons are displayed on the screen.
Valid Value	{0 1} 0 –Disable 1 - Enable
Default Value	0

6 WORKFLOW EXAMPLES

This section provides examples of the workflow of a typical HD system during basic operations. By following the numbered order of the commands, screen text, and actions presented in the tables, you can understand the action/effect relationship among the application interface, the screen console, and the HD device.

In the following examples:

- Commands listed in the Application column are entered into the end-user application.
- Text that appears on the screen, resulting either from the entry of command or an action by the HD device, appears in the Console column.
- Actions that occur in the HD device are described in the HD column.

6.1 Incoming Call Flow

The following table presents the flow of events and actions that result in a successful incoming videoconference call:

APPLICATION	CONSOLE	HD
1. Connect to console	2. Welcome to HD <model number>Version Board Type Camera Input	3. System completes the initialization phase.
	4. EVENT_SYSTEM_INIT_COMPLETED_SUCCESS	
5. CallEvents 1	6. CALLEVENTS OK	7. Incoming call.
	8. EVENT_CALL_OFFERING 0 10.11.24 Demo	
9. CallAnswer 1	10. CALLANSWER OK	11. Call is connected successfully.
	12. EVENT_CALL_CONNECTED 0	13. Outgoing video channel opened.

	14. EVENT_CALL_CHANNEL_ON 0 0 TX VIDEO	15. Outgoing audio channel opened.
	16. EVENT_CALL_CHANNEL_ON 0 1 TX AUDIO	17. Outgoing data channel opened.
	18. EVENT_CALL_CHANNEL_ON 0 2 TX DATA	19. Incoming Video channel opened.
	20. EVENT_CALL_CHANNEL_ON 0 3 RX VIDEO	21. Incoming Audio channel opened.
	22. EVENT_CALL_CHANNEL_ON 0 4 RX AUDIO	23. Incoming Data channel opened.
	24. EVENT_CALL_CHANNEL_ON 0 5 RX DATA	
25. Hangup 0	26. HANGUP OK	27. Outgoing video channel closed.
	28. EVENT_CALL_CHANNEL_OF F 0 0 TX VIDEO	29. Outgoing audio channel closed.
	30. EVENT_CALL_CHANNEL_OF F 0 1 TX AUDIO	31. Outgoing data channel closed.
	32. EVENT_CALL_CHANNEL_OF F 0 2 TX DATA	33. Incoming Video channel closed.
	34. EVENT_CALL_CHANNEL_OF F 0 3 RX VIDEO	35. Incoming Audio channel closed.
	36. EVENT_CALL_CHANNEL_OF F 0 4 RX AUDIO	37. Incoming Data channel closed.
	38. EVENT_CALL_CHANNEL_OF F 0 5 RX DATA	39. Call is disconnected.
	40. EVENT_CALL_DISCONNECT ED 0 LOCAL	41. System becomes idle.
	42. EVENT_CALL_IDLE 0	

6.2 Outgoing Call Flow

The following table presents the flow of events and actions that result in a successful outgoing videoconference call:

APPLICATION	CONSOLE	HD
1. Connect to console	2. Welcome to HD <model number>Version Board Type Camera Input	3. System completes the initialization phase.
	4. <code>EVENT_SYSTEM_INIT_COMPLETED_SUCCESS</code>	
5. CallEvents 1	6. CALLEVENTS OK	
7. Dial 10.0.11.12	8. DIAL OK	9. Call state dialtone arrived
	10. <code>EVENT_CALL_DIALTONE 0</code>	11. Call state ringback arrived
	12. <code>EVENT_CALL_RINGBACK 0</code>	13. Call become connected
	14. <code>EVENT_CALL_CONNECTED 0</code>	15. Outgoing video channel opened
	16. <code>EVENT_CALL_CHANNEL_ON 0 0 TX VIDEO</code>	17. Outgoing audio channel opened.
	18. <code>EVENT_CALL_CHANNEL_ON 0 1 TX AUDIO</code>	19. Outgoing data channel opened.
	20. <code>EVENT_CALL_CHANNEL_ON 0 2 TX DATA</code>	21. Incoming Video channel opened.

	22. <code>EVENT_CALL_CHANNEL_ON 0 3 RX VIDEO</code>	23. Incoming Audio channel opened.
	24. <code>EVENT_CALL_CHANNEL_ON 0 4 RX AUDIO</code>	25. Incoming Data channel opened.
	26. <code>EVENT_CALL_CHANNEL_ON 0 5 RX DATA</code>	27. Remote party disconnects the call
	28.	29. Outgoing video channel closed.
	30. <code>EVENT_CALL_CHANNEL_OFF 0 0 TX VIDEO</code>	31. Outgoing audio channel closed.
	32. <code>EVENT_CALL_CHANNEL_OFF 0 1 TX AUDIO</code>	33. Outgoing data channel closed.
	34. <code>EVENT_CALL_CHANNEL_OFF 0 2 TX DATA</code>	35. Incoming Video channel closed.
	36. <code>EVENT_CALL_CHANNEL_OFF 0 3 RX VIDEO</code>	37. Incoming Audio channel closed.
	38. <code>EVENT_CALL_CHANNEL_OFF 0 4 RX AUDIO</code>	39. Incoming Data channel closed.
	40. <code>EVENT_CALL_CHANNEL_OFF 0 5 RX DATA</code>	41. Call is disconnected.
	42. <code>EVENT_CALL_DISCONNECTED D 0 REMOTE</code>	43. Call become idle
	44. <code>EVENT_CALL_IDLE 0</code>	