

# Phoenix

**External Control Protocol  
Reference Manual  
020-101425-05**



# Phoenix

**External Control Protocol  
Reference Manual  
020-101425-05**

## NOTICES

### COPYRIGHT AND TRADEMARKS

© 2016 Christie Digital Systems USA Inc. All rights reserved.

All brand names and product names are trademarks, registered trademarks or trade names of their respective holders.

### REGULATORY

The product has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the product is operated in a commercial environment. The product generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of the product in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at the user's own expense.

CAN ICES-3 (A) / NMB-3 (A)

이 기기는 업무용(A급)으로 전자파적합등록을 한 기기이오니 판매자 또는 사용자는 이점을 주의하시기 바라며, 가정 외의 지역에서 사용하는 것을 목적으로 합니다.

### GENERAL

Every effort has been made to ensure accuracy, however in some cases changes in the products or availability could occur which may not be reflected in this document. Christie reserves the right to make changes to specifications at any time without notice. Performance specifications are typical, but may vary depending on conditions beyond Christie's control such as maintenance of the product in proper working conditions. Performance specifications are based on information available at the time of printing. Christie makes no warranty of any kind with regard to this material, including, but not limited to, implied warranties of fitness for a particular purpose. Christie will not be liable for errors contained herein or for incidental or consequential damages in connection with the performance or use of this material.

The product is designed and manufactured with high-quality materials and components that can be recycled and reused. This symbol means that electrical and electronic equipment, at their end-of-life, should be disposed of separately from regular waste. Please dispose of the product appropriately and according to local regulations. In the European Union, there are separate collection systems for used electrical and electronic products. Please help us to conserve the environment we live in!



### GENERAL WARRANTY STATEMENTS

For complete information about Christie's limited warranty, please contact your Christie dealer. In addition to the other limitations that may be specified in Christie's limited warranty, the warranty does not cover:

- a. Problems or damage occurring during shipment, in either direction.
- b. Problems or damage caused by combination of a Product with non-Christie equipment, such as distribution systems, cameras, video tape recorders, etc., or use of a Product with any non-Christie interface device.
- c. Problems or damage caused by misuse, improper power source, accident, fire, flood, lightening, earthquake or other natural disaster.
- d. Problems or damage caused by improper installation/alignment, or by equipment modification, if by other than Christie service personnel or a Christie authorized repair service provider.
- e. Problems or damage caused by use of a Product on a motion platform or other movable device where such Product has not been designed, modified or approved by Christie for such use.
- f. Problems or damage caused by use of a system in the presence of an oil-based fog machine.
- g. Failure due to normal wear and tear.

### PREVENTATIVE MAINTENANCE

Preventative maintenance is an important part of the continued and proper operation of your product. Please see the Maintenance section for specific maintenance items as they relate to your product. Failure to perform maintenance as required, and in accordance with the maintenance schedule specified by Christie, will void the warranty.

# Contents

Getting Connected .....	7
Ethernet Connectivity .....	7
Logon .....	7
Commands and Responses .....	8
Command Overview Table .....	8
Command Responses .....	9
Supported Command Descriptions .....	9
Logon .....	10
Logoff .....	10
LayerAdd .....	10
LayerResetAspect .....	11
LayoutRecall .....	11
LayerRemove .....	12
LayerMove .....	12
GetLayer .....	13
GetLayers .....	13
GetLayouts .....	14
GetSource .....	14
GetSources .....	18
SourceApply .....	19
ResetSource .....	19
SetAudioSource .....	20
GetNodes .....	20
GetUser .....	21
GetUsers .....	22
GetWall .....	22
GetWalls .....	23
GetTreatments .....	23
TreatmentApply .....	24
Restart .....	24
Shutdown .....	24
Index .....	26

# Examples

1: Logon Example .....	10
------------------------	----

---

2: Logoff Example.....	10
3: Layer Add Example .....	11
4: Reset Layer Aspect Example.....	11
5: Layout Recall Example.....	11
6: Layer Remove Example .....	12
7: Layer Move Example 1 - Absolute Positioning .....	12
8: Layer Move Example 2 - Relative Positioning .....	12
9: Get Layer Example.....	13
10: Get Layers Example.....	14
11: Get Layouts Example.....	14
12: Get Source [RTSP Stream] Example .....	16
13: Get Source [Still Image] Example .....	16
14: Get Source [VNC] Example.....	17
15: Get Source [Remote Desktop] Example.....	17
16: Get Source [Application] Example .....	18
17: Get Source [Phoenix DVI Input] Example .....	18
18: Get Sources Example.....	19
19: Source Apply Example .....	19
20: ResetSource Example .....	19
21: Set Audio Source Example.....	20
22: Get Nodes Example .....	20
23: Get User Example .....	21
24: Get Users Example .....	22
25: Get Wall Example.....	22
26: Get Walls Example .....	23
27: Get Treatments Examples .....	23
28: Treatment Apply Example .....	24
29: Restart Example.....	24
30: Shutdown Example.....	25

# Getting Connected

The Phoenix node uses an ASCII based command system for external control. Connectivity is available via an RJ-45 (Ethernet) connection, using a human readable (ASCII) command/response set.

Each command sent to the controller must be terminated with a carriage return character, and the commands and arguments must be separated by a space character. All responses from the controller will be terminated with a '>' character.

## Ethernet Connectivity

The Phoenix node can be controlled remotely by sending the ASCII commands listed later within this document by first opening a TCP socket connection to port 11135 on the frame.

## Logon

Upon establishing connection, the first command sent by the client should be a logon command. The logon command authenticates a supplied username / password against a previously created Phoenix user account, and sets the permission level for subsequent commands (a 'NotAuthorized' response will be returned for commands if the logged on user lacks sufficient permissions).

### Data Formats

The format for the command argument and response parts are:

//TODO: Fill in Date/Time, Number, and Text

# Commands and Responses

This section contains the command arguments and responses in the required format.

## Command Overview Table

The following commands are available when using an Ethernet connection to a Phoenix controller.

Command	Description	Page Reference
<b>GetLayer</b>	Gets information on a single layer on a wall.	13
<b>GetLayers</b>	Gets a list of layers on a wall.	13
<b>GetLayouts</b>	Gets a list of all layouts.	14
<b>GetNodes</b>	Gets a list of all nodes.	20
<b>GetSource</b>	Gets information on a single source.	13
<b>GetSources</b>	Gets a list of all sources.	13
<b>GetTreatments</b>	Gets a list of all treatments.	23
<b>GetUser</b>	Gets information on a single user.	19
<b>GetUsers</b>	Gets a list of all users.	19
<b>GetWall</b>	Gets information on a single wall.	22
<b>GetWalls</b>	Gets a list of all walls.	13
<b>LayerAdd</b>	Adds a new layer to a wall.	10
<b>LayerMove</b>	Moves the position of a layer.	12
<b>LayerRemove</b>	Removes one or more layers from a Wall.	11
<b>LayerResetAspect</b>	Resets the aspect ratio of a layer.	11
<b>LayoutRecall</b>	Recalls an existing layout.	11
<b>Logoff</b>	Logs user off.	10
<b>Logon</b>	Logs on a user.	10
<b>ResetSource</b>	Resets the decoders that are consuming the source.	19
<b>Restart</b>	Restarts all the nodes connected to the controller.	24
<b>SetAudioSource</b>	Sets the audio source for a wall.	24



Command	Description	Page Reference
<b>Shutdown</b>	Powers off all the nodes assigned to the controller.	24
<b>SourceApply</b>	Applies a source to a layer.	19
<b>TreatmentApply</b>	Applies a treatment to a layer.	24

**Notes:** Commands are not case sensitive.

Some commands display their last argument as 'Argument x ~ XX:'. These commands allow the last argument field to be repeated, allowing multiples of the last argument to be specified with a single command.

Some commands display argument(s) with an asterisk. These arguments are optional; however no further arguments may be entered in the command.

String argument spaces must be replaced with the three character ASCII string %20 to create a valid command, as space is used as the argument delimiter.

When used as part of a response, the '>' character will be replaced with the four character ASCII string &gt; since this is the character used as a response terminator.

Commands sent from the client must be terminated with a carriage return character. The external controller response will be terminated with two carriage returns and a > character.

## Command Responses

A response is returned for every command sent to the Phoenix controller. If multiple values are returned in a single response, the argument delimiter is an ASCII space character, and argument values containing a space will be converted to the three character ASCII string "%20". The first response argument is always the result code for the command, which lets you know if a command was successfully processed. The table below displays the various responses that can be returned by the Phoenix controller.

Response	Description
Success	Command was successfully processed.
Empty	Data requested is not available.
InvalidCommand	Invalid command specified.
InvalidArgCount	Missing required minimum number of arguments.
InvalidArgValue	One or more arguments were invalid.
Execution	An execution error occurred while processing the command.
NotAuthorized	No user is logged on, or the user who is logged on does not have the permission level necessary to run the command.

## Supported Command Descriptions

This section describes the details for each command.

## Logon

Associated a user to the current external control connection by sending a username / password to the controller.

Command: Logon

Argument 1: *[Text]* UserName

Argument 2: *[Text]* Password

Response 1: *[Number]* Result code, with one of the following values:

- 0: Success
- 1: Invalid Username / Password
- 2: Password Expired
- 3: Unknown / Other Failure

### 1: Logon Example

<b>Command Description</b>	Logon as Joe, with a password of Test12345
<b>Command</b>	Logon Joe Test12345
<b>Response</b>	Success 0>

## Logoff

Removes any previously logged in user from the current external control session.

Command: Logoff

### 2: Logoff Example

<b>Command Description</b>	Log off the currently logged on user
<b>Command</b>	Logoff
<b>Response</b>	Success>

## LayerAdd

Adds a new layer to a wall, specifying the associated source, size, and position for the new layer.

Command: LayerAdd

Argument 1: *[Number]* Wall ID to create the new layer on.

Argument 2: *[Number]* Source ID for the new layer.

Argument 3: *[Number]* X Position, in pixels, relative to the left edge of the wall.

Argument 4: *[Number]* Y Position, in pixels, relative to the top edge of the wall.

Argument 5: *[Number]* Width, in pixels, for the new layer

Response 1: *[Number]* Resulting layer ID created, or -1 if no layer was created

Response 2: *[Number]* Result code, with one of the following values:

- 0: Success
- 1: Failed - Unknown Error occurred
- 2: Failed - Too many layers would be created on the output
- 3: Failed – Unable to start a required transcode operation for the specified source

**3: Layer Add Example**

<b>Command Description</b>	Add source with ID 3 as a new layer onto the wall with ID 0 and an X,Y of 330,500 and a width of 800 pixels
<b>Command</b>	LayerAdd 0 3 330 500 800
<b>Response</b>	Success 14 0>
<b>Response Description</b>	Layer ID 14 was created successfully

## LayerResetAspect

Resets the aspect ratio of a layer.

*Command:* LayerResetAspect

*Argument 1:* *[Number]* Layer ID

Response 1: Success

Response 2: Failure:

**4: Reset Layer Aspect Example**

<b>Command Description</b>	Resets the aspect ratio of a layer
<b>Command</b>	LayerResetAspect 1
<b>Response</b>	Success >
<b>Response Description</b>	The aspect ratio for Layer ID 1 was reset successfully

## LayoutRecall

Recalls a provided layout ID on the preset’s associated wall(s).



After using this command, use the GetLayers command to get the current layers on the wall.

*Command:* LayoutRecall

*Argument 1:* *[Number]* Preset ID to recall

*Response:* <No Response Parameters>

**5: Layout Recall Example**

<b>Command Description</b>	Recall Layout ID 25
<b>Command</b>	LayoutRecall 25
<b>Response</b>	Success>

## LayerRemove

Removes one or more layer(s) from one or more walls.

Command: LayerRemove

Argument 1-XX: *[Number]* Layer ID

### 6: Layer Remove Example

<b>Command Description</b>	Remove layer IDs 5, 8, and 16 from their various walls
<b>Command</b>	LayerRemove 5 8 16
<b>Response</b>	Success>

## LayerMove

Moves the position of one or more layers to either an absolute pixel location on a wall, or offsets the current layer position by a horizontal and/or vertical offset.

Command: LayerMove

Argument 1: *[Number]* Move Type (0 = Absolute Position, 1 = Offset Position)

Argument 2: *[Number]* Move Directions (H = Horizontal, V = Vertical, B = Both)

Argument 3: *[Number]* Horizontal Offset in Pixels (Not used but **required** if move direction is V)

Argument 4: *[Number]* Vertical Offset in Pixels (Not used but **required** if move direction is H)

Argument 5-XX: *[Number]* Layer ID(s) to move

Response 1-XX: Result code for each layer specified in the command, taking the form:

<Layer ID>:<Result Code>. Possible result codes are listed below:

- 0: Success
- 1: Move would have created too many layers on a single output
- 2: Unknown Error processing request

### 7: Layer Move Example 1 - Absolute Positioning

<b>Command Description</b>	Move Layer ID 10 to position 200, 100 on its wall
<b>Command</b>	LayerMove 0 B 200 100 10
<b>Response</b>	Success 10:0>

### 8: Layer Move Example 2 - Relative Positioning

<b>Command Description</b>	Move Layer IDs 12 and 13 up 50 pixels on their respective walls
<b>Command</b>	LayerMove 1 V 0 -50 12 13
<b>Response</b>	Success 12:0 13:0>

## GetLayer

Gets information on a single layer on a wall.

Command: GetLayer

Argument 1: *[Number]* Wall ID

Argument 2: *[Number]* Layer ID

Response 1: *[Number]* Layer ID

Response 2: *[Number]* Wall ID

Response 3: *[Number]* Source ID

Response 4: *[Number]* z index

Response 5: *[Number]* absolute top

Response 6: *[Number]* absolute left

Response 7: *[Number]* Width

Response 8: *[Number]* Height

### 9: Get Layer Example

<b>Command Description</b>	Gets information on a single layer on a wall
<b>Command</b>	GetLayer 0 0
<b>Response</b>	Success 0 0 5 0 12 100 693 433 >

## GetLayers

Gets a list of layers on a wall.

Command: GetLayers

Argument 1: *[Number]* Wall ID

Response 1: *[Number]* Number of layer on the wall

Response 2: *[Number]* Layer ID

Response 3: *[Number]* Wall ID

Response 4: *[Number]* Source ID

Response 5: *[Number]* z index

Response 6: *[Number]* absolute top

Response 7: *[Number]* absolute left

Response 8: *[Number]* Width

Response 9: *[Number]* Height

Response X: *[Loop]* Responses 2-9 loop for each wall in the response

### 10: Get Layers Example

<b>Command Description</b>	Gets a list of layers on a wall.
<b>Command</b>	GetLayers 0
<b>Response</b>	Success 2 0 0 5 0 12 100 693 433 1 0 3 1 247 849 960 540 >

## GetLayouts

Returns a brief listing of layouts (Presets) defined in the system.

Command: GetLayouts

<No Arguments Required>

Response 1: *[Number]* Number of layouts in the system being returned

Response 2: *[Number]* Layout ID

Response 3: *[Text]* Layout Name

Response X: *[Loop]* Responses 2-3 loop for each layout in the response

### 11: Get Layouts Example

<b>Command Description</b>	Get a listing of all layouts in the system. Two layouts will be returned: <ul style="list-style-type: none"> <li>- Layout ID 0 ("Main Look")</li> <li>- Layout ID 1 ("PC1 FullScreen")</li> </ul>
<b>Command</b>	GetLayouts
<b>Response</b>	Success 2 0 Main%20Look 1 PC1%20FullScreen>

## GetSource

Returns detailed information about a source. The first part of the response is always the same but depending on the type the remaining responses will change.

Command: GetSource

Argument 1: *[Number]* Source ID to get information on

Response 1: *[Number]* Source ID

Response 2: *[Text]* Source Name

Response 3: *[Text]* Title

Response 4: *[Number]* Width

Response 5: *[Number]* Height

Response 6: *[Number]* Aspect Ratio

Response 7: *[Number]* Default Treatment ID, or -1 if no Default Treatment is set

Response 8: *[Number]* Type ID of the Source

- 0 - RTSP Stream

- 1 - Still Image
- 2 - VNC
- 3 - Remote Desktop
- 4 – Application
- 5 - Phoenix DVI Input

**RTSP Stream**

Response 9: *[Text]* IP Address

Response 10: *[Number]* Port

Response 11: *[Text]* Video File Name

Response 12: *[Number]* Keep Alive Interval

Response 13: *[Number]* Audio Available (0 = False, 1 = True)

Response 14: *[Number]* Force Unicast (0 = False, 1 = True)

**Still Image**

Response 9: *[Text]* File Name

**VNC**

Response 9: *[Text]* IP Address

Response 10: *[Number]* Port

Response 11: *[Number]* Display Index

Response 12: *[Number]* Encode Bit Rate

Response 13: *[Number]* Reachback Available (0 = False, 1 = True)

Response 14: *[Number]* Is Manned (0 = False, 1 = True)

**Remote Desktop**

Response 9: *[Text]* IP Address

Response 10: *[Number]* Port

Response 11: *[Text]* User Name

Response 12: *[Number]* Encode Bit Rate

Response 13: *[Number]* Audio Available (0 = False, 1 = True)

Response 14: *[Number]* Reachback Available (0 = False, 1 = True)

Response 15: *[Number]* Is Manned (0 = False, 1 = True)

**Application**

Response 9: *[Text]* Command

**Phoenix DVI Input**

Response 9: *[Text]* Encoder Node Mac Address

Response 10: *[Text]* Input Channel (A = Input A, B = Input B)

Response 11: *[Number]* Encode Bit Rate

Response 12: *[Number]* Audio Available (0 = False, 1 = True)

Response 13: *[Number]* Reachback Available (0 = False, 1 = True)

Response 14: *[Number]* Is Manned (0 = False, 1 = True)

**12: Get Source [RTSP Stream] Example**

<b>Command Description</b>	<p>Get details of a RTSP Stream Source with the following values:</p> <ul style="list-style-type: none"> <li>- ID: 8</li> <li>- Name: Axis 1</li> <li>- Title: Office Camera</li> <li>- Width: 1920</li> <li>- Height: 1080</li> <li>- Aspect Ratio: 16 x 9</li> <li>- Default Treatment: -1</li> <li>- Type: RTSP Stream</li> <li>- IP Address: 10.10.30.25</li> <li>- Port: 554</li> <li>- Video File Name: axis-media/media.amp?videocodec=h264&amp;audio=1</li> <li>- Keep Alive Interval: 30</li> <li>- Audio Available: True</li> <li>- Force Unicast: False</li> </ul>
<b>Command</b>	GetSource 8
<b>Response</b>	Success 8 Axis%201 Office%20Camera 1920 1080 1.7778 -1 0 10.10.30.25 554 axis-media/media.amp?videocodec=h264&audio=1 30 1 0 >

**13: Get Source [Still Image] Example**

<b>Command Description</b>	<p>Get details of a Still Image Source with the following values:</p> <ul style="list-style-type: none"> <li>- ID: 32</li> <li>- Name: Jellyfish</li> <li>- Title: Box Jellyfish</li> <li>- Width: 1024</li> <li>- Height: 768</li> <li>- Aspect Ratio: 4 x 3</li> <li>- Default Treatment: 1</li> <li>- Type: Still Image</li> <li>- File Name: Jellyfish.jpg</li> </ul>
<b>Command</b>	GetSource 32
<b>Response</b>	Success 32 Jellyfish Box%20Jellyfish 1024 768 1.3333 1 1 Jellyfish.jpg >



**14: Get Source [VNC] Example**

<b>Command Description</b>	<p>Get details of a VNC Source with the following values:</p> <ul style="list-style-type: none"> <li>- ID: 9</li> <li>- Name: Windows 7 VNC</li> <li>- Title: Test Machine 2</li> <li>- Width: 1024</li> <li>- Height: 768</li> <li>- Aspect Ratio: 4 x 3</li> <li>- Default Treatment: -1</li> <li>- Type: VNC</li> <li>- IP Address: 10.10.30.250</li> <li>- Port: 5900</li> <li>- Display Index: 0</li> <li>- Encode Bit Rate: 4000</li> <li>- Reachback Available: True</li> <li>- Is Manned: False</li> </ul>
<b>Command</b>	GetSource 9
<b>Response</b>	Success 9 Windows%207%20VNC Test%20Machine%202 1024 768 1.3333 -1 2 10.10.30.250 5900 0 4000 1 0 >

**15: Get Source [Remote Desktop] Example**

<b>Command Description</b>	<p>Get details of a Remote Desktop Source with the following values:</p> <ul style="list-style-type: none"> <li>- ID: 1</li> <li>- Name: Windows 7 RDP</li> <li>- Title: Test Machine</li> <li>- Width: 1920</li> <li>- Height: 1080</li> <li>- Aspect Ratio: 16 x 9</li> <li>- Default Treatment: -1</li> <li>- Type: Remote Desktop</li> <li>- IP Address: 10.10.30.249</li> <li>- Port: 3389</li> <li>- User Name: Eric</li> <li>- Encode Bit Rate: 4000</li> <li>- Audio Available: False</li> <li>- Reachback Available: True</li> <li>- Is Manned: False</li> </ul>
<b>Command</b>	GetSource 1
<b>Response</b>	Success 1 Windows%207%20RDP Test%20Machine 1920 1080 1.7778 -1 3 10.10.30.249 3389 Eric 4000 0 1 0 >

**16: Get Source [Application] Example**

<b>Command Description</b>	Get details of an Application Source with the following values: <ul style="list-style-type: none"> <li>- ID: 21</li> <li>- Name: Test Application</li> <li>- Title: Microsoft Website</li> <li>- Width: 1920</li> <li>- Height: 1080</li> <li>- Aspect Ratio: 16 x 9</li> <li>- Default Treatment: -1</li> <li>- Type: Web</li> <li>- Website: www.microsoft.com</li> </ul>
<b>Command</b>	GetSource 21
<b>Response</b>	Success 21 Test%20Application Microsoft%20Website 1920 1080 1.7778 -1 4 www.microsoft.com >

**17: Get Source [Phoenix DVI Input] Example**

<b>Command Description</b>	Get details of a Phoenix DVI Input Source with the following values: <ul style="list-style-type: none"> <li>- ID: 31</li> <li>- Name: Test</li> <li>- Title: Desktop</li> <li>- Width: 1920</li> <li>- Height: 1080</li> <li>- Aspect Ratio: 16 x 9</li> <li>- Default Treatment: -1</li> <li>- Type: Phoenix DVI Input</li> <li>- Encoder Node Mac Address: E8-40-F2-D2-37-EF</li> <li>- Input Channel: A</li> <li>- Encode Bit Rate: 4000</li> <li>- Audio Available: False</li> <li>- Reachback Available: False</li> <li>- Is Manned: False</li> </ul>
<b>Command</b>	GetSource 31
<b>Response</b>	Success 31 Test Desktop 1920 1080 1.7778 -1 5 E8-40-F2-D2-37-EF A 4000 0 0 0 >

## GetSources

Returns a brief listing of sources defined in the system.

Command: GetSources

<No Arguments Required>

Response 1: *[Number]* Number of sources in the system being returned

Response 2: *[Number]* Source ID

Response 3: *[Text]* Source Name

Response X: *[Loop]* Responses 2-3 loop for each source in the response

**18: Get Sources Example**

<b>Command Description</b>	Get a listing of all sources in the system. Two sources will be returned: <ul style="list-style-type: none"> <li>- Source ID 0 ("Blu-Ray 1")</li> <li>- Source ID 1 ("PC 1")</li> </ul>
<b>Command</b>	GetSources
<b>Response</b>	Success 2 0 Blu-Ray%201 1 PC%201>

## SourceApply

Applies a source to a layer.

Command: SourceApply

Argument 1: *[Number]* Layer ID

Argument 2: *[Number]* Source ID

Response 1: *Success*

Response 2: Failed - Unknown Error occurred or invalid layer or source ID

**19: Source Apply Example**

<b>Command Description</b>	Applies a source to a layer
<b>Command</b>	SourceApply 0 2
<b>Response</b>	Success >
<b>Response Description</b>	Applied source 2 to layer 0

## ResetSource

Resets all decoders that are currently using the source.

Command: ResetSource

Argument 1: *[Number]* Source ID to reset

**20: ResetSource Example**

<b>Command Description</b>	Resets the source on all decoders
<b>Command</b>	ResetSource 1
<b>Response</b>	Success>

## SetAudioSource

Sets the audio source for a wall.

Command: SetAudioSource

Argument 1: *[Number]* Wall ID

Argument 2: *[Number]* Source ID, use -1 to set no audio source

Response: <No Response Parameters>

An InvalidArgValue may occur for the following reasons:

- Wall does not exist
- Source does not exist
- Source does not support audio
- Source does not allow audio

### 21: Set Audio Source Example

<b>Command Description</b>	Set Audio Source for Wall 0 with Source 8
<b>Command</b>	SetAudioSource 0 8
<b>Response</b>	Success>

## GetNodes

Returns a brief listing of nodes defined in the system.

Command: GetNodes

<No Arguments Required>

Response 1: *[Number]* Number of nodes in the system being returned

Response 2: *[Text]* Node serial number

Response 3: *[Text]* Node MAC address

Response X: *[Loop]* Responses 2-3 loop for each node in the response

### 22: Get Nodes Example

<b>Command Description</b>	Get a listing of all nodes in the system. Two nodes will be returned: <ul style="list-style-type: none"> <li>- Number of nodes</li> <li>- Node serial number and MAC Address("03FF28 00094803FF28")</li> <li>- Node serial number and MAC Address("00008B E840F2D29140")</li> </ul>
<b>Command</b>	GetNodes
<b>Response</b>	Success 2 3FF28 00094803FF28 00008B E840F2D29140>

## GetUser

Returns detailed information on a specified user ID.

Command: GetUser

Argument 1: *[Number]* User ID to get information on

Response 1: *[Number]* User ID

Response 2: *[Text]* UserName

Response 3: *[Text]* First Name

Response 4: *[Text]* Last Name

Response 5: *[Text]* Company Name

Response 6: *[Date/Time]* Last Logon Time

Response 7: *[Text]* Notes associated with the user account

Response 8: *[Text]* Image File name for user account

Response 9: *[Number]* Email Address Count

Response 10: *[Text]* Email Address Type (Home / Work)

Response 11: *[Text]* Email Address

Response X: *[Loop]* Response 10-11 repeated for the number count provided in response 9

Response 12: *[Number]* Contact Number Count

Response 13: *[Text]* Contact Number Type (Home / Cell / Work)

Response 14: *[Text]* Contact Number

Response X: *[Loop]* Response line 13-14 loop for each contact type (# indicated in Response 10)

### 23: Get User Example

<b>Command Description</b>	Get details on User ID 7, which has the following parameters: <ul style="list-style-type: none"> <li>- User ID: 7</li> <li>- UserName: jsmith</li> <li>- First Name: Joe</li> <li>- Last Name: Smith</li> <li>- Company: AV Integrators, Inc.</li> <li>- Last Logon Time: February 21, 2013 8:32AM</li> <li>- Notes: Integrator service account</li> <li>- Image File: jsmith-thumbnail.jpg</li> <li>- Cell Phone: 555-123-4567</li> <li>- Home Phone: 555-123-8223</li> <li>- Work Email: <a href="mailto:joe.smith@avintegratorsinc.com">joe.smith@avintegratorsinc.com</a></li> <li>- Home Email: joe.smith@outlook.com</li> </ul>
<b>Command</b>	GetUser 1
<b>Response</b>	Success 7 jsmith Joe Smith AV%20Integrators,%20Inc. 01/21/13%208:32 Integrator%20service%20account jsmith-thumbnail.jpg 2 Work <a href="mailto:joe.smith@avintegratorsinc.com">joe.smith@avintegratorsinc.com</a> Home joe.smith@outlook.com 2 Cell 555-123-4567 Home 555-123-8223>

## GetUsers

Returns a brief listing of all users defined in the system.

Command: GetUsers

<No Arguments Required>

Response 1: *[Number]* Number of users in the system being returned

Response 2: *[Number]* User ID

Response 3: *[Text]* UserName

Response 4: *[Text]* First Name

Response 5: *[Text]* Last Name

Response X: *[Loop]* Responses 2-5 loop for each user in the response

### 24: Get Users Example

<b>Command Description</b>	Get a listing of all users in the system. Two users will be returned: <ul style="list-style-type: none"> <li>- User ID 0 ("jsmith") Joe Smith</li> <li>- User ID 8 ("tjamison") Tom Jamison</li> </ul>
<b>Command</b>	GetUsers
<b>Response</b>	Success 2 jsmith Joe Smith tjamison Tom Jamison>

## GetWall

Returns detailed configuration information about a specified wall ID.

Command: GetWall

Argument 1: *[Number]* Wall ID

Response 1: *[Number]* Wall ID

Response 2: *[Text]* Wall Name

Response 3: *[Number]* Width in pixels

Response 4: *[Number]* Height in pixels

Response 5: *[Number]* Background Color – Red (0-255)

Response 6: *[Number]* Background Color – Green (0-255)

Response 7: *[Number]* Background Color – Blue (0-255)

Response 8: *[Text]* Background Image Name

Response 9: *[Text]* Background Image Horizontal Alignment (L=Left, C=Center, R=Right)

Response 10: *[Text]* Background Image Vertical Alignment (T=Top, M=Middle, B=Bottom)

Response 11: *[Text]* Background Stretch (N=None, F=Fill, U=Uniform, UF=Uniform Fill, T=Tile)

Response 12: *[Number]* Audio Source ID, or -1 for no audio source

### 25: Get Wall Example

<b>Command Description</b>	Get information on wall ID 3, which has the following parameters: <ul style="list-style-type: none"> <li>- Name: Main Wall</li> <li>- Size: 3840 x 1024</li> </ul>
----------------------------	--

	<ul style="list-style-type: none"> <li>- Background Color: Black</li> <li>- Background Image: LogoBG.jpg</li> <li>- Audio Source: 4</li> </ul>
Command	GetWall 3
Response	Success 3 Main%20Wall 3840 1024 0 0 0 LogoBG.jpg 4>

## GetWalls

Returns a brief listing of walls defined in the system.

Command: GetWalls

<No Arguments Required>

Response 1: *[Number]* Number of walls in the system being returned

Response 2: *[Number]* Wall ID

Response 3: *[Text]* Wall Name

Response X: *[Loop]* Responses 2-3 loop for each wall in the response

### 26: Get Walls Example

Command Description	Get a listing of all walls in the system. Two walls will be returned: <ul style="list-style-type: none"> <li>- Wall ID 0 ("Wall 1") width of 3840 and a height of 1024</li> <li>- Wall ID 1 ("Wall 2") width of 2560 and a height of 1600</li> </ul>
Command	GetWalls
Response	Success 2 0 0 Wall%201 3840 1024 1 Wall%202 2560 1600>

## GetTreatments

Returns a brief listing of treatments defined in the system.

Command: GetTreatments

<No Arguments Required>

Response 1: *[Number]* Number of treatments in the system being returned

Response 2: *[Number]* Treatment ID

Response 3: *[Text]* Treatment Name

Response X: *[Loop]* Responses 2-3 loop for each treatment in the response

### 27: Get Treatments Examples

Command Description	Get a listing of all treatments in the system. Two treatments will be returned: <ul style="list-style-type: none"> <li>- Treatment ID 0 ("Red Border")</li> <li>- Treatment ID 1 ("Letterbox Crop")</li> </ul>
Command	GetTreatments

Response	Success 2 0 Red%20Border 1 Letterbox%20Crop>
----------	--

## TreatmentApply

Applies a treatment to a layer.

Command: TreatmentApply

Argument 1: *[Number]* Treatment ID

Argument 2: *[Number]* Layer ID2

Argument 3: *[Number]* Layer ID3

Argument 4: *[Number]* Layer ID4

Argument 5: *[Number]* Layer IDN

Response 1: *Success*

Response 2: Failed - Unknown Error occurred

### 28: Treatment Apply Example

Command Description	Applies a treatment to one or more layers
Command	TreatmentApply 1 0
Response	Success >
Response Description	Applied treatment 1 to layer 0

## Restart

Restarts all nodes that are currently connected to the controller.

Command: Restart

### 29: Restart Example

Command Description	Restarts all nodes connected to the controller
Command	Restart
Response	Success>

## Shutdown

Shutdown (powers off) all nodes that are currently connected to the controller.

Command: Shutdown



**30: Shutdown Example**

<b>Command Description</b>	Shutdown all nodes connected to the controller
<b>Command</b>	Shutdown
<b>Response</b>	Success>

# Index

GetLayer .....	13	LayerMove .....	12
GetLayers .....	13	LayerRemove.....	12
GetLayouts.....	14	LayerResetAspect.....	11
GetNodes .....	20	LayoutRecall.....	11
GetSource .....	14	logoff.....	10
GetSources .....	18	Logon .....	10
GetTreatments.....	23	ResetSource .....	19
GetUser .....	21	Restart .....	24
GetUsers.....	22	SetAudioSource .....	20
GetWall.....	22	Shutdown.....	24
GetWalls .....	23	SourceApply .....	19
LayerAdd .....	10	TreatmentApply .....	24

#### Corporate offices

USA – Cypress  
ph: 714-236-8610

Canada – Kitchener  
ph: 519-744-8005

#### Consultant offices

Italy  
ph: +39 (0) 2 9902 1161

#### Worldwide offices

Australia  
ph: +61 (0) 7 3624 4888

Brazil  
ph: +55 (11) 2548 4753

China (Beijing)  
ph: +86 10 6561 0240

China (Shanghai)  
ph: +86 21 6278 7708

Eastern Europe and  
Russian Federation  
ph: +36 (0) 1 47 48 100

France  
ph: +33 (0) 1 41 21 44 04

Germany  
ph: +49 2161 664540

India  
ph: +91 (080) 6708 9999

Japan (Tokyo)  
ph: 81 3 3599 7481

Korea (Seoul)  
ph: +82 2 702 1601

Republic of South Africa  
ph: +27 (0)11 510 0094

Singapore  
ph: +65 6877-8737

Spain  
ph: +34 91 633 9990

United Arab Emirates  
ph: +971 4 3206688

United Kingdom  
ph: +44 (0) 118 977 8000