ZUMMESH-AVBRIDGE
AV Bridge
Installation Guide

Description

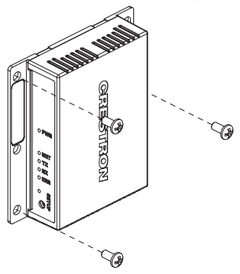
The Crestron® ZUMMESH-AVBRIDGE is an AV bridge that facilitates communication between a Zūm™ lighting system and a third-party AV system. The ZUMMESH-AVBRIDGE interfaces with the AV system using an RS-232 or USB cable. Without the need for a ZUM-FLOOR-HUB, the AV bridge converts a serial protocol to Zūm commands to, for example, recall a scene, disable an occupancy sensor, or change the room occupancy state.

Additional Resources

Visit the product page on the Crestron website (www.crestron.com) for additional information and the latest firmware updates. Use a QR reader application on your mobile device to scan the QR image.

Install the ZUMMESH-AVBRIDGE

The ZUMMESH-AVBRIDGE can be mounted onto any flat surface using three screws that are appropriate for the mounting surface (not included). For best results, position the device to avoid interference from nearby RF sources, obstructions, and metal surfaces.

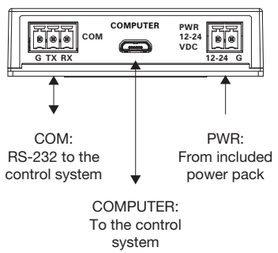


Wire the ZUMMESH-AVBRIDGE

To wire the ZUMMESH-AVBRIDGE, first make the power and communications connections.

The AV Bridge can be powered from a 12-24Vdc power supply or via the USB connection. Make the connections from the ZUMMESH-AVBRIDGE to the AV system. For details, refer to the illustration that follows.

NOTE: Use the COM or COMPUTER port to make the connection to the AV system. Both connections cannot be used at the same time. If both connections are made, the ZUMMESH-AVBRIDGE will use USB to send commands.



The details for the COM port are listed below.

G: Ground TX: Transmit RX: Receive

Default serial protocol speed: 115.2k baud (8 data bits, 1 stop bits, no flow control, no parity).

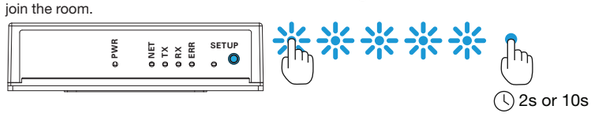
Creating a Zūm Commercial Lighting System

NOTE: This can be performed on only one device in the room.

NOTE: The device that is used to create the room is automatically added to the room. The device does not need to be added to the room.

NOTE: A room can be created only from an ac-powered device.

To start a new Zūm Commercial Lighting System, press the **SETUP** button 5 times, and then press and hold the **SETUP** button for about 2 seconds. The LED will light. If the device is not factory fresh, hold the button for about 10 seconds. The LED illuminates for 3 seconds and then slowly flashes to indicate that the room is in Joining mode and that other devices can join the room.



Add the ZUMMESH-AVBRIDGE to a Zūm System

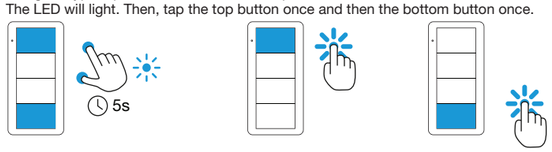
To add the ZUMMESH-AVBRIDGE to a Zūm system, first place the system in Joining mode, then add the ZUMMESH-AVBRIDGE to the system, and then exit Joining mode.

Enter Joining Mode

NOTE: The LEDs on all ac-powered devices in the system flash when the system is in Joining mode.

To enter Joining mode, do the following:

- Using a keypad, press and hold both the top and bottom buttons for about 5 seconds. The LED will light. Then, tap the top button once and then the bottom button once.



- Using a keypad or j-box device, press the **SETUP** button 2 times, and then press the **TEST** button.

NOTE: Joining mode is automatically entered after creating a new system. For details, refer to the Zūm Setup Guide (Doc. 7957) at www.crestron.com/manuals.

NOTE: Press any button on a device that is part of the system to exit Joining mode. Joining mode exits automatically after 4 minutes.

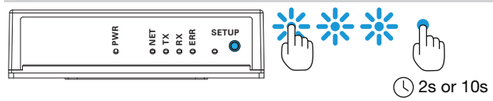
Add the ZUMMESH-AVBRIDGE to the System

NOTE: A Zūm mesh device can belong to only one room.

NOTE: The Zūm mesh device used to create the room is already part of the network. It does not need to be added to the network.

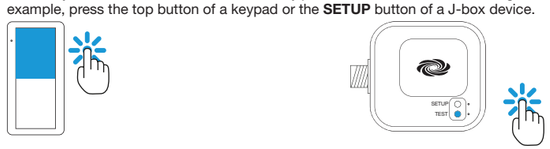
To add the ZUMMESH-AVBRIDGE to the system, press the **SETUP** button 3 times, and then press and hold it for about 2 seconds. The LED will light. Release the button when the LED lights. The LED flashes slowly to indicate that it is part of the room and that the room is still in Joining Mode.

NOTE: If the device is not factory fresh, hold the button for about 10 seconds. The LED will light.



Exit Joining Mode

Press any button on a device that has already joined the network to exit Joining mode. For example, press the top button of a keypad or the **SETUP** button of a J-box device.



AV Bridge Command List

NOTE: All commands and responses are case insensitive.

AV Bridge Serial Protocol and Syntax

Data packets are carriage-return delimited (0xd) ASCII text (lines).

NOTE: All text is case insensitive.

- Special Characters:
  - Sent to the AV Bridge:
    - ! Command: sets a value or triggers an action (e.g., lights off, recall scene, etc.)
    - ? Request: used to query a value or state of a Zūm device (e.g., the dimmer level)
    - $ Pass-through command: used to invoke debug commands
  - Sent from the AV Bridge:
    - ~ Response: sent in response to a command (!) or request (?)
    - ^ Feedback: sent when a condition in the room changes (e.g., scene change, occupancy change, button press, etc.)
    - \ line continuation: sent at end of a line when multiple responses are issued
- Line feeds (0x0a) are ignored.
- White space is not permitted in commands, except for pass-through commands.
- After sending a packet to the AVB, characters sent before the AVB returns the "prompt" will be discarded. The prompt is "AVB>".

The AV Bridge responds to commands with the prompt "AVB>". If a command has a syntax or value error, an error message is sent before the prompt. When a command changes the room's state, it may trigger a feedback message. The Filter command may prevent certain commands from being displayed, depending on the filter level setting. Refer to the Filter command for details on feedback filtering.

AV Bridge IDs

To address specific room devices, the AV bridge communicates directly with the devices using an AV Bridge ID (AVID). The AVID consists of a K or L, to identify the category of the device as a keypad or load controller, and a two-digit number that is assigned for the device. The AVID can be, for example, L01, L02, K01, K02, K03, etc.

NOTE: The leading 0 is not needed when referencing an AVID; for example, L1 can be written as "L1" or "L01".

NOTE: In the commands that follow, "[id]" would be substituted with an AVID.

Error Responses

Error messages can be received from the AV system or the AV Bridge for the following reasons.

ERROR TYPE	ERROR RESPONSE	ERROR PRIORITY	COMMENTS
Command error	~error	1	Command not recognized; blank lines not treated as an error
Parameter error	~command.err.param	2	Missing or too many parameters passed or an invalid AVID referenced (e.g., invalid category or two-digit number)
Non-existent device referenced	~command.err.no-exist	3	"command": any command referencing a device by AVID
Parameter range error	~command.err.range	4	Error with parameter value other than AVID

Feedback Messages

After a command is received by the AV Bridge, a feedback message may be generated. A feedback message is preceded by a ^ and is issued to acknowledge that the room changed state. Feedback messages may be sent when an AV system sends a command, a button is pressed in the room, or an occupancy sensor detects occupancy. Feedback messages are sent to the AV control system and must not get mixed with response or prompt text.

NOTE: A Zūm Floor Hub and BLE phone app can also cause state changes.

Administrator Commands

Command Line Pass-Thru Command

Lines that start with "$" will be passed to the AV bridge's command line processor to allow low-level debugging of the system.

Debug Command

The Debug command sets the debug control state. The debug state can also be queried. When the debug state changes, the feedback response is issued by the AV Bridge.

Debug Control Command

Command Format	Command Response	Query Format	Query Response
!debug.[enable|disable]	AVB>	?debug	~debug.[enabled|disabled]

Enable debugging:

```
AVB> !debug.enable
AVB>
```

Echo Command

The Echo command enables or disables the device echo. The echo state can also be queried. When the echo state changes, the feedback response is issued by the AV Bridge.

Echo Command

Command Format	Command Response	Query Format	Query Response
!echo.[enable|disable]	AVB>	?echo	~echo.[enabled|disabled]
Notes:	Determines whether the device local echos the incoming character stream		

Turn off Echo:

```
AVB> !echo.enable
AVB>
```

Fw (Firmware) Command

The Fw query reports the version of the firmware.

Firmware Version Command

Query Format	Query Response
?fw	~fw.[version]

Query the firmware version:

```
AVB> ?fw
~fw.xxxyyyzzzz
AVB>
```

Reset Command

The Reset command resets the AV Bridge. When the AV Bridge is reset, a feedback response is issued by the AV Bridge.

Reset AV Bridge Command

Command Format	Command Response
!reset	AVB>
Notes:	This will reset the AV bridge. The AV bridge will respond with the ~reset confirmation before resetting.

Reset the AV Bridge:

```
AVB> !reset
AVB>
```

SetAVID Command

The SetAVID pass-through command assigns an AVID to a device. This can be used to change an AVID when a device is replaced, or it can be used to implement preplanned AVID assignments based on the device's serial number. The -a parameter allows an AVID to be changed to another AVID. The -s parameter allows an AVID to be assigned to a device based on the device's serial number.

SetAVID Command

Command Format	Command Response
$SetAVID -a [AVID] [AVID] $SetAVID - s [SN] [AVID]	AVB>
Notes:	Permits reassignment of an AVID

Replace AVID L01 with L03:

```
AVB> $SetAVID -a [L01] [L03]
AVB>
```

Assign AVID L04 to serial number 123456789:

```
AVB> $SetAVID -s [L04] [123456789]
AVB>
```

DeleteDev Command

The DeleteDev pass-through command deletes a device from the AV Bridge. This can be used to free up an AVID for use by another device, or it can be used to remove a device from the system. The -a parameter allows an AVID to be removed using an AVID. The -s parameter allows an AVID to be removed from a device based on serial number.

DeleteDEV Command

Command Format	Command Response
$DeleteDEV -a [AVID] $DeleteDEV -s [SN]	AVB>
Notes:	Permits deleting an AVID. -s can be used only when in Master mode (i.e., no net bridge in the room).

Delete AVID L01:

```
AVB> $DeleteDEV -a [L01]
AVB>
```

Delete the AVID from a device with the serial number 123456789:

```
AVB> $DeleteDEV -s [123456789]
AVB>
```

Zūm Network Management Functions

Permitjoin Command

The Permitjoin command places the room into Joining mode. The room's permitjoin state can also be queried.

Enable/Disable Joining Command

Command Format	Command Response	Query Format	Query Response
!permitjoin.[enable|disable]	AVB>	?permitjoin	~permitjoin.[enable|disable]
Notes:	Joining mode automatically times out after 4 minutes.		

Enter Joining mode:

```
AVB> !permitjoin.enable
AVB>
```

Form Command

The Form command creates a new Zūm network.

Form Network Command

Command Format	Command Response
!form	AVB>
Notes:	The device leaves any network and forms a new network. The response should be immediate. The form command puts the network in permit joining mode. It can use Permit Join command to disable joining immediately. If not, permit join times out.

Form a new network:

```
AVB> !form
AVB>
```

Identify Command

The Identify command identifies ac-powered Zūm devices that are part of the network by flashing their LED.

Identify Device Command

Command Format	Command Response
!identify.[id|stop]	AVB>
Notes:	Times out after 3 minutes The stop parameter stops

Identify load controller with AVID L01:

```
AVB> !identify.L01
AVB>
```

Join Command

The Join command tells a device to join a Zūm network that is in Joining mode.

Join Network Command

Command Format	Command Response	Query Format	Query Response
!join	AVB>	?join	~join.[success|failure]
Notes:	If the device was commanded to join a network, the ~join response appears only after a success or failure. If a query was issued, the response should be immediate.		

Join a network that is in Joining mode:

```
AVB> !join
~join.success
AVB>
```

Request the join state of the device:

```
AVB> ?join
~join.success
AVB>
```

Button Command

The Button response is sent in response to a button press.

Button Actions Command

Command Format	Command Response	Feedback Response
!button.[tap|hold|release].[button].[id].[SN]	AVB>	^button.[tap|hold|release].[button].[id].[SN]
Notes:	Button values: tap, hold, release SN: serial number string	

When the AV Bridge receives information about a button 1 hold from a keypad with AVID K01, the AV Bridge sends the following response:

```
AVB> ^button.hold.1.K01.123456789
```

Level Command

The Level command sets the load controller level. The load controller level can also be queried. When the load controller's state changes, the feedback response is issued by the AV Bridge.

Direct Dimmer Levels Command

Command Format	Command Response	Query Format	Query Response	Feedback Response Format (filter-level == 2)
!level.[level].[id] !level.[level]	AVB>	?level.[id] Cannot request all dimmer levels at once	^level.[level].[id]	~level.[level].[id]
Notes:	Level is percentage of max brightness (0-100), with or without leading zeros. There is no fade time associated with this command.			Sent in response to !level.[level].[id], !level.[level] or when changed from within room.

Establish a 50% load level for the load controller assigned to AVID L01:

```
AVB> !level.50.L01
AVB>
AVB> ^level.50.L01
```

Set all load controllers to 100 (when there are two load controllers in the room—AVID L01 and L02)?

```
AVB> !level.100
AVB>
AVB> ^level.100.L01
AVB> ^level.100.LO2
```

Request a load level of the load controller assigned to AVID L02:

```
AVB> ?level.L02
~level.74.L02
AVB>
```

Filter Command

The Filter command sets the feedback message filtering. The filtering state can also be queried.

Filter Command

Command Format	Command Response	Query Format	Query Response
!filter.[0|1|2]	AVB>	?filter	~filter.[0|1|2]
Notes:	Feedback messages sent for each setting: 0: Only button action messages (default) 1: Button actions and room-level state changes 2: Button actions, room level, and device level state changes		

Change the feedback message filtering:

```
AVB> !filter.2
AVB>
```

Dev Command

The Dev command forms a list of all devices in the room.

List Devices Command

Query Format	Query Response
?dev	~dev.[Device ID].[status].[SN] or ~dev.none
Notes:	Notes: Repeats the response for each device with an AVID. Load controllers follow the form Lxx and keypads follow Kxx. For each device an individual response line is sent. All but the last response have a line continuation character "\" at the end of the line. If no devices exist, response is ~dev.none. Status: "active" or "missing" SN: Serial number string

List all devices on the Zum network:

```
AVB> ?dev
~dev.L01.active.123456789\
~dev.K01.active.213456789\
~dev.L02.missing.312456789
AVB>
```

Plug Command

The Plug command turns the plug load controller on or off. The plug load controller state can also be queried. When the plug load controller's state changes, the feedback response is issued by the AV Bridge.

Plug Load Control / Feedback Command

Command Format	Command Response	Query Format	Query Response	Feedback Response Format (filter-level == 2)
!plug.[on|off]	AVB>	?plug	~plug.[on|off]	^plug.[on|off]
Notes:	Plug load controllers cannot be individually controlled. The feedback is OR'd in case there are multiple plug controllers that get out of sync.			Sent in response to !plug.[on|off] or if change sent from within room

Turn on the plug load controllers:

```
AVB> !plug.on
AVB>
AVB> ^plug.on
```

Ramp Command

The Ramp command increases or decreases the dimmer levels. When the ramp state changes, the feedback response is issued by the AV Bridge.

Ramp - Raise/Lower/Stop Lights Command

Command Format	Command Response
!ramp.[up|down|stop] !ramp.[up|down|stop].[id]	AVB>
Notes:	The lights ramp using an up or down command, and then stop with a stop command. The ramp stops based on the device timeout setting or if a stop command is received. When ramping down, lights stop at their minimum level and do not turn off.

Ramp up all load controllers. To prevent ramping to 100, issue the ramp stop command:

```
AVB> !ramp.up
AVB>
AVB> !ramp.stop
AVB>
```

Scenesave Command

The Scenesave command saves the current light levels as a scene.

Scenesave Command

Command Format	Values	Feedback Response
!scenesave.[number]	1-16	~scenesave.[number]

Save the current light levels for scene 5:

```
AVB> !scenesave.5
AVB>
```

Sync Command

The Sync command queries the system to verify that an AV Bridge is present. When the AV Bridge is booted up, a feedback response is issued by the AV Bridge.

Sync Command

Query Format	Query Response	Feedback Response
?sync	~sync	^sync
Notes:	This allows the AV control system to verify that the AV bridge is up and running.	Sent after boot when AV bridge is ready to process AV commands

Query the sync status:

```
AVB> ?sync
~sync
AVB>
```

Room-Level Commands

Room-level commands send command and request commands to an entire room. Responses are sent based on the state of the room.

Command Format	Query Format	Query Response	Feedback Response Format (filter-level >=1)
!room.[Field].[value]	?room.[Field]	~room.[Field].[value]	^room.[Field].[value]

Scene Command

The Scene command sets the room lights to their scene levels. The room's current scene can also be queried. When the room's scene changes, the feedback response is issued by the AV Bridge.

Scene Command

Field	Values	Query Format	Query Response	Feedback Response
Scene	1-16	?room.scene	~room.Scene.10	^room.Scene.10

Set a room-level scene to scene 1:

```
AVB> !room.scene.1
AVB>
AVB> ^room.scene.1
```

Request the current scene in the room:

```
AVB> ?room.scene
~room.scene.#
AVB>
```

Occupancy Command

The Occupancy command sets the room occupancy state for the room. The room's occupancy state can also be queried. When the room's occupancy state changes, the feedback response is issued by the AV Bridge.

Occupancy Command

Field	Values	Query Format	Query Response	Feedback Response
Occupancy	Occupied, Vacant	?room.occupancy	~room.occupancy.[occupied|vacant]	^room.occupancy.[occupied|vacant]

Set the room as vacant:

```
AVB> !room.occupancy.vacant
AVB>
AVB> ^room.occupancy.vacant
```

Occ-action Command

The Occ-action command enables or disables the occupancy action state for the room. The room's occupancy action status can also be queried. When the room occupancy action status changes, the feedback response is issued by the AV Bridge.

Occ-action Command

Field	Values	Query Format	Query Response	Feedback Response
Occ-action	Enable, Disable	?room.occ-action	~room.occ-action.[enable|disable]	^room.occ-action.[enable|disable]

Disable occupancy actions:

```
AVB> !room.occ-action.disable
AVB>
AVB> ^room.occ-action.disable
```

Query occupancy action status:

```
AVB> ?room.occ-action
~room.lights.off
AVB>
```

Lights Command

The Lights command toggles the lights for the room. The room's light state can also be queried. When the room light status changes, the feedback response is issued by the AV Bridge.

Lights Command

Field	Values	Query Format	Query Response	Feedback Response
Lights	On, Off	?room.lights	~room.lights.[on|off]	^room.lights.[on|off]

Enable occupancy actions:

```
AVB> !room.lights.on
AVB>
AVB> ^room.lights.on
```

Query lights status:

```
AVB> ?room.lights
~room.lights.on
AVB>
```

Photo Command

The room's photocell value can be queried. When the room's photocell value changes, the feedback response is issued by the AV Bridge.

Photo Command

Field	Values	Query Format	Query Response	Feedback Response
Photo	0-65535	?room.photo	~room.photo.[0-65535]	^room.photo.[0-65535]

Query photocell level:

```
AVB> ?room.photo
~room.photo.6813
AVB>
```

Daylight-Action Command

The Daylight-Action command enables or disables daylighting for the room. The room's daylight-action value can also be queried. When the room's daylight-action value changes, the feedback response is issued by the AV Bridge.

Daylight-action Command

Field	Values	Query Format	Query Response	Feedback Response
Daylight-Action	Enable, Disable	?room.daylight-action	~room.daylight-action.[enable|disable]	^room.daylight-action.[enable|disable]

Enable daylight action:

```
AVB> !room.daylight-action.enable
AVB>
AVB> ^room.daylight-action.enable
```

Query daylight-action status:

```
AVB> ?room.daylight-action
~room.daylight-action.enable
AVB>
```


Crestron Electronics, Inc.
15 Volvo Drive, Rockleigh, NJ 07647
Tel: 888.CRESTRON
Fax: 201.767.7576
www.crestron.com

Installation Guide - DOC. 7916B
(2047921)
11.17
Specifications subject to change without notice.