

**Partner: BiAmp**  
**Model: AudiaFlex & Nexia**  
**Device Type: DSP**



## GENERAL INFORMATION

<b>SIMPLWINDOWS NAME:</b>	Biamp AudiaFlex + Nexia Level Control v5.1
<b>CATEGORY:</b>	Mixer
<b>VERSION:</b>	V5.1
<b>SUMMARY:</b>	This module controls any level point in the BiAmp AudiaFlex or Nexia.
<b>GENERAL NOTES:</b>	<p>This module will control any level point in the Biamp AudiaFlex and Nexia.</p> <p>This module <b>MUST</b> be used in conjunction with the Biamp AudiaFlex + Nexia Command Processor v5.1.umc module. This module processes all transmitted and received serial strings and reformats device feedback so that this data can be sent to the proper module for final processing.</p> <p>When polling the BiAmp for current status, you should poll for only the information you really need at the time. The more data points you poll for at one time, the longer it will take to get an update for any one data point. It should not normally be necessary to poll for all data points all the time.</p> <p>This module has (7) seven parameter fields, all of which must be set for proper module operation. All parameters are entered as ASCII characters. Volume_Device_Type is the control block type. This selected from a drop down list. Volume_Device_ID is the device's ID and is automatically assigned when the .dap file is compiled. Volume_Device_Instance is the "Logic Block's" ID that is automatically assigned when the .dap file is compiled. Volume_Index_1 is the first index number from the BiAmp software. This is typically the channel, input or output number to be controlled. Volume_Index_2 is the second index number from the BiAmp software. In a lot of cases this will be zero. Volume_Upper_Limit is the volume level's upper limit. This should be entered as the dB level and negative numbers are allowed. Volume_Lower_Limit is the volume level's lower limit. This should be entered as the dB level and negative numbers are allowed.</p> <p>This information is all contained in the Block properties field when developing the .dap file within the Biamp AudiaFlex Windows software. A .dap file (Crestron Test v5.dap) was created by Crestron for testing purposes and <b>MUST</b> be used for proper operation of the Pro2 DEMO v5.1 program.</p> <p><b>NOTE: THIS MODULE WAS DEVELOPED AND TESTED WITH THE BIAMP AUDIAFLEX. THE INCLUDED .DAP FILE WAS PROVIDED BY BIAMP, AND IS FOR THE AUDIAFLEX ONLY. ACCORDING TO BIAMP, THESE MODULES WILL WORK FOR THE NEXIA. A CONFIGURATION FILE WILL NEED TO BE CREATED FOR THE NEXIA (NOT PROVIDED), AND WILL BE REQUIRED FOR OPERATION OF THE UNIT. FOR MORE INFORMATION ABOUT CONFIGURATION FILES AND HOW TO CREATE THEM PLEASE CONTACT BIAMP.</b></p> <p>All responses from the BiAmp must be routed through the BiAmp AudiaFlex + Nexia Unit Buffer v5.1 module. This module will send the response string to only modules that are controlling the particular instance in the BiAmp. If there are more than 20 modules controlling a single instance object in the BiAmp, you must add buffering outside this module to send the response to no more than 20 modules at a time. Please see the demo program for an example of this.</p>
<b>CRESTRON HARDWARE REQUIRED:</b>	ST-COM, C2-COM

**Partner: BiAmp**  
**Model: AudiaFlex & Nexia**  
**Device Type: DSP**



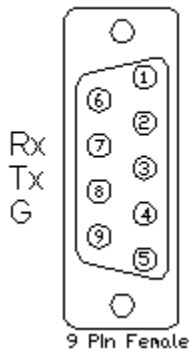
## SETUP OF CRESTRON HARDWARE:

RS232  
 Baud: 38400  
 Parity: N  
 Data Bits: 8  
 Stop Bits: 1

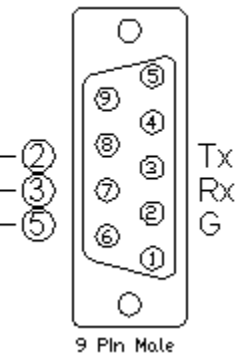
## VENDOR FIRMWARE:

4.380

Rear View of Connector



Rear View of Connector



## CONTROL:

<b>Volume_Up/Down</b>	D	Press and hold to adjust the volume level.
<b>New_Volume_Level</b>	A	Analog value of volume level. This is the signed dB level to set the volume level to. Will be sent when the Send_New_Level input is pulsed. This will allow preset values to be sent to the BiAmp.
<b>Send_New_Level</b>	D	Pulse to send the volume entered in the New_Volume_Level input. This will allow preset values to be sent to the BiAmp.
<b>Poll_Level</b>	D	Pulse to poll for the current value.
<b>From_Processor\$</b>	S	Serial data signal coming from the Biamp AudiaFlex + Nexia Unit Buffer v5.1 module.

**Partner: BiAmp**  
**Model: AudiaFlex & Nexia**  
**Device Type: DSP**

**FEEDBACK:**

<b>Volume_Bar</b>	A	Analog value indicating the current volume level. To be displayed using a bar graph on a touch panel.
<b>Volume_Level_Sign_Unscaled</b>	A	Analog volume level value. This is the signed dB level. To be displayed using a digital gauge on a touch panel.
<b>To_Processor\$</b>	S	Serial data signal to be sent to the Biamp AudiaFlex + Nexia Command Processor v5.1.

**PARAMETERS:**

<b>Volume_Device_Type</b>	ASCII	Select the proper device type from the drop down list.
<b>Volume_Device_ID</b>	ASCII	Device address automatically assigned after the Biamp .dap file is compiled
<b>Volume_Device_Instance</b>	ASCII	Logic Block ID assigned after the Biamp .dap file is compiled
<b>Volume_Index_1</b>	ASCII	Volume index to be controlled. This is the input, channel or output number being controlled.
<b>Volume_Index_2</b>	ASCII	This used for cross point type devices. Typically, this would be zero. For cross points this is the output number for the cross point being controlled.
<b>Volume_Upper_Limit</b>	ASCII	This is the upper limit for the volume level being controlled. This is the signed dB value.
<b>Volume_Lower_Limit</b>	ASCII	This is the lower limit for the volume level being controlled. This is the signed dB value.

**Partner: BiAmp**  
**Model: AudiaFlex & Nexia**  
**Device Type: DSP**

**TESTING:****OPS USED FOR TESTING:**

3.155.1240

**SIMPL WINDOWS USED FOR TESTING:**

2.08.44

**CRES DB USED FOR TESTING:**

18.09.02.001

**SYMBOL LIBRARY USED FOR TESTING:**

531

**SAMPLE PROGRAM:**

BiAmp AudiaFlex + Nexia Demo Pro2 v5.1

**REVISION HISTORY:**

V3 – 2-Series Only, corrected dialer timing, text display, speed of dialing and over all operation (firmware)

V4 – Changed timing of dialer strings sent when off hook

V5 – Made changes for the new responses from the BiAmp. These new responses have the command details and status in them. This eliminates the need to poll for status when making changes. Added new commands. Added buffering for the responses to improve system response.

V5.1-Changed the Command Processor module to handle the response for presets. Also eliminated the Command Processor sending any response if the unit ID is determined to be 0. Changed all of the modules to allow instance IDs up to 65534d. Changed all modules to look for the proper channel ID. Added MBMUTE command to the On-Off module.