AudioInRoom

KNX controller for audio with Bluetooth and auxiliary input – 4 outputs

ZMU-AUIR

Application Program Version: 1.3
User manual edition: [1.3]_a

CONTENTS

DOCUMENT UPDATES

Version	Changes	Page(s)
[1.3]_a	**Changes in the application programme:** • Internal optimisation.	-
[1.2]_a	**Changes in the application programme:** • Pairing Time is increased up to 3 minutes. • Sending status objects after power tension. • New track information objects. • Setting auxiliary input or Bluetooth streaming as the priority source for playback control. • Two additional tones: Welcome and Wake Up (only supported from serial number 20ACC0494 on). • Room State Control: "No dot disturb" for muting the Ring and Welcome Tones. • Ckech-in/out protection improved.	-
[1.1]_a	**Changes in the application programme:** • New pairing PIN generation modes. • Resetting the volume value when switching audio source device. • Check-in/out: process is not interrupted when receiving a new order. • New parameter to select the polarity of the mute object. • New object to indicate the pairing mode status.	-

1. INTRODUCTION

1.1 AUDIOINROOM

AudioInRoom from Zennio is KNX interface capable of playing audio from an external device, such as a smartphone, which is connected via Bluetooth or through an auxiliary input (for example, for televisions).

The most notable features of this device are:

- **Audio Playback** from an external user device.

- **Audio Available Controls:**

  - ➢ Play/Stop.

  - ➢ Next track/Previous track.

  - ➢ Volume.

- Track Information.

- **Bluetooth/Auxiliary Input** audio source selector.

- Bluetooth pairing with configurable **password**.

- **Ring tone.**

- ⚠ Two additional tones: **Welcome** and **Wake Up**.

- **Do not Disturb** for Ring Tone and Ring Wake Up.

- **2 independent stereo channels of audio output.**

- **Heartbeat** or periodic "still-alive" notification.

⚠ **Important Warning**: *Functionality marked in this document with a warning icon is only supported from serial number 20ACC0494 on.*

1.2 AUDIO PLAYBACK SYSTEM

The following figure shows the main elements involved in the audio playback system through AudioInRoom, either from an external Bluetooth source or from the Auxiliary Input:
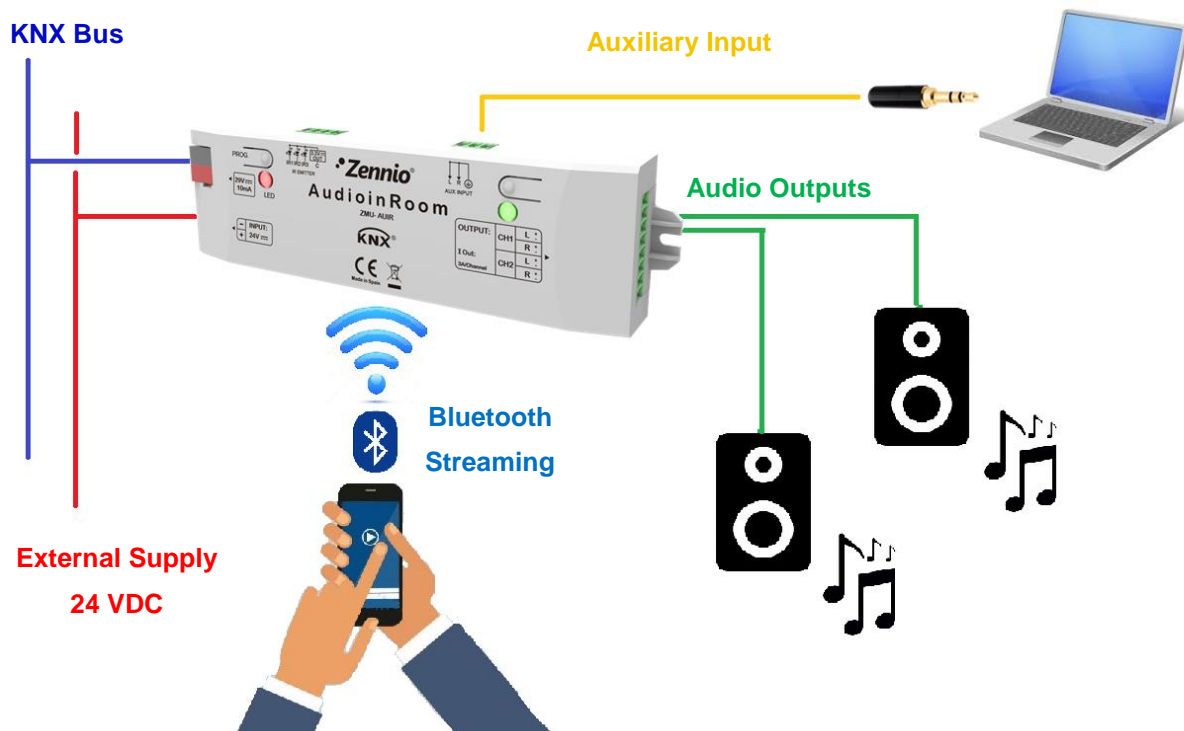

Figure 1. AudioInRoom playback system elements.

- **AudioInRoom**: KNX system for audio playback from a device connected via Bluetooth or through the auxiliary input. Enables Play/Stop, Next track/Previous track and volume control through KNX bus.

- **Bluetooth Streaming**: Bluetooth audio source linked to the KNX device by means of a pairing with configurable password.

- **Auxiliary Input:** AudioInRoom can also be connected to systems without Bluetooth. This is possible thanks to the Auxiliary Input, using an analogue audio connector (mini Jack) or a RCA connector. In these cases, the playback control can only be carried out from the own user device audio player.

● **Audio Outputs:** two independent channels with two connectors each for speakers, which can be enabled or disabled individually. The audio output signal from each channel can be configured to be played in stereo or mono mode.

1.3 INSTALLATION

AudioInRoom is connected to the KNX bus through the built-in terminal (2). This device requires an **external** 24V DC **power supply** (1).

1. 24VDC Power Supply.
2. KNX Connector.
3. Programming LED.
4. Programming Button.
5. Not used.
6. Audio Auxiliary Input.
7. Ventilation Grill.
8. Testing Button.
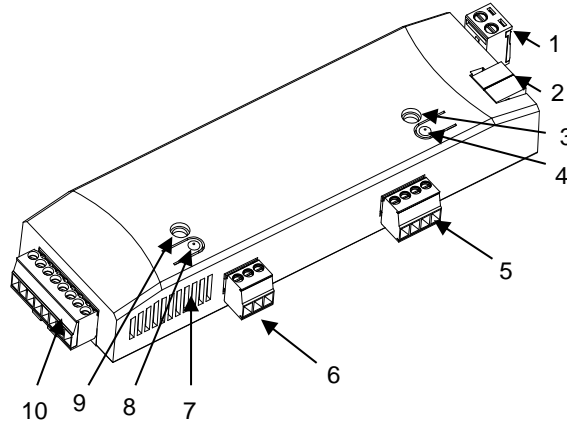9. Testing LED.
10. Audio Output Channels.

Figure 2. AudioInRoom.

A short press in **Programming Button** (4) will make the device enter the programming mode. The **Programming LED** (3) will then light in red. On the contrary, if this button is held while the device gets connected to the bus, AudioInRoom will enter the **safe mode**. In such case, the programming LED will blink in red colour.

Moreover, a 3 seconds press in **Testing Button** (8) turns the Bluetooth module on and sets the device into **pairing mode**. The **Testing LED** (9) will light:

- In blue when the Bluetooth module is on.

- In green when pairing is active.

- Flashes red in the absence of KNX bus voltage, blending with the other colours if several notifications are made at the same time.

If a device without a Bluetooth module is to be connected, it must be connected via the **audio auxiliary input** (6). Figure 3 shows the correct AudioInRoom auxiliary input connection, depending on whether a Jack analogue audio connector or an RCA connector is used:
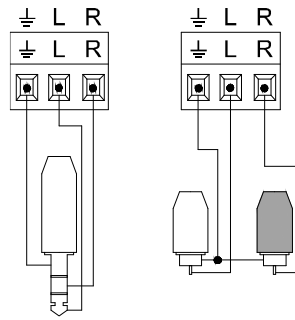
Figure 3. Auxiliary Input: Jack connector (left) and RCA connector (right).

Once the AudioInRoom is connected to an audio source, the processed audio output will take place through the **audio output channels** (10). Two independent channels with two connectors each for the speakers are available.

The following figure shows how to connect the audio output channels:
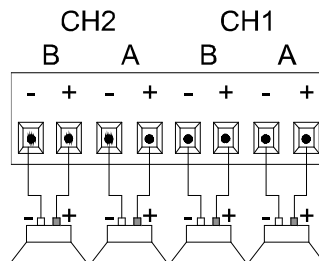

Figure 4. Audio Output Channels.

For detailed information about the technical features of the device and for safety instructions or about the installation process, please refer to the corresponding **Datasheet**, bundled with the original package of the device and also available at www.zennio.com.

1.4 BLUETOOTH CONNECTION

AudioInRoom connection with an audio source device through the Bluetooth module is a simple procedure that maintains connection security at all times when carried out using a configurable password.

1.4.1 CHECK-IN / CHECK-OUT

The' Check-in' process represents the arrival of a new user to the room and establishes a new identifier name for the AudioInRoom and a new password or PIN to link to it. These two values are set by communication object.

Instead, the Check-out process resets the name and password to the default (parameter) settings. In addition, the list of devices linked to AudioInRoom is deleted, so that any user who wants to reconnect to AudioInRoom must repeat the pairing process.

The Check-in or Check-out processes will not be interrupted by the arrival of new Check-in or Check-out orders. Once finished, the last order received is analysed and, if it is the opposite than the process just finished, the pertinent actions will be executed.

1.4.2 PAIRING

The pairing process is initiated in AudioInRoom by sending the order via the corresponding communication object or by pressing (more than three seconds) the test button.

Once the pairing mode is activated, the AudioInRoom will remain visible via Bluetooth for approximately 3 minutes ⚠️. It can be identified by the name established in the Check-In process or by the name established by parameter in ETS.

⚠️ *For serial numbers prior to 20ACC0494 the time it will remain visible will be 1 minute.*

When selecting the desired AudioInRoom to connect to, **the pairing PIN is requested**. This PIN, again, may be the one established in the Check-in process or the one established by parameter in ETS.

After entering the PIN, the audio source device and AudioInRoom will be paired and connected:

- **Connected Device:** the device has already been paired through the security PIN and it is connected to AudioInRoom. At this point the sound from the device will be heard through the speakers connected to the AudioInRoom audio output channels. In addition, playback and volume controls can be used from the AudioInRoom

- **Paired Device**: the device has already been paired through the security PIN. However, the audio-multimedia connection has been disabled. AudioInRoom does not allow the user to play music in this state.

AudioInRoom enables to have up to two devices paired and connected, so that audio can be played from any of them, although, logically, not at the same time. When the device playing the audio changes, the volume of the AudioInRoom is reset to the initial value set by parameter.

If a third device is paired, one of the two previous devices will be deleted, which means that it will have to be paired again if it wants to reconnect.

Note: *If a pairing process starts while a Bluetooth device is playing music, and it is not successful (cancellation, incorrect PIN or time exceeded), the playback will pause.*

1.5 INICIALIZATION AND POWER TENSION

After a programming, a *Check-out* procedure is executed, which will also update the Bluetooth module with PIN and device name values set by parameter.

With a power failure, the Bluetooth connection gets lost. This causes the source device playback gets stopped. AudioInRoom has been designed so that audio source playback can be recovered in a known status.

On the other hand, a KNX bus power failure **does not imply any change in the device status since it has external power supply**. The Testing LED will red flashing for information.

After power tension, the following status objects are sent to the KNX bus:

- [Audio] On/Off (Status)

- [Audio] Check-in/Check-out (Status)

- [Audio][BT] Name (Status)

- [Audio] Aux. Input (Status)

- [Audio] Channel X: Mute (Status)

- [Audio] Channel X, Subchannel A: Mute (Status)

- [Audio] Channel X, Subchannel B: Mute (Status)

- [Audio] Channel X: Volume Limit (Status)

- [Audio] Channel X: Volume Limit Value (Status)

- [Audio] Channel X: Volume (Status)

- [Audio] Channel X, Subchannel A: Volume (Status)

- [Audio] Channel X, Subchannel B: Volume (Status)

- [Audio] Channel X: Alarm (Status)

- [Audio] Alarm: Overheating (Status)

- [Audio] Title of the Track

- [Audio] Name of the Artist

2. CONFIGURATION

After importing the corresponding database in ETS and adding the device into the topology of the desired project, the configuration process begins by entering Parameters tab of the device.

2.1 GENERAL

From this screen the two main functions of AudioInRoom, **Heartbeat** and **Bluetooth Module**, can be enabled.

ETS PARAMETERISATION

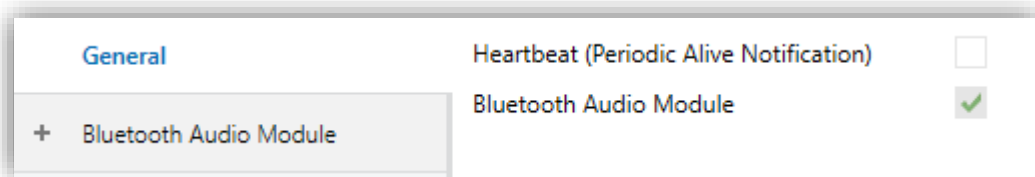The configurable parameters on the General screen are the following:

Figure 5. General Configuration.

- **Heartbeat (Periodic Alive Notification)** [*disabled/enabled*][1]: incorporates a one-bit object to the project ("**[Heartbeat] Object to Send '1'**") that will be sent periodically with a value of "1" to notify that the device is still working (still *alive*).

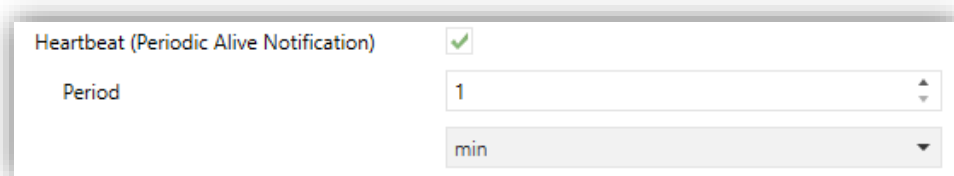**Figure 6.** Heartbeat (Periodic Alive Notification).

> **Note**: *The first sending after download or bus failure takes place with a delay of up to 255 seconds, according to the individual address, to prevent bus overload. The following sendings match the period set.*

[1] The default values of each parameter will be highlighted in blue in this document, as follows: [*default/rest of options*].

- **Bluetooth Audio Module** [*enabled*]: Bluetooth module is the main functionality of the AudioInRoom and is enabled permanently. For more information, see section 2.2.

2.2 BLUETOOTH AUDIO MODULE

All aspects of the Bluetooth module are configured from this tab:

- **Device Name:** customizable name for easy AudioInRoom identification while searching and pairing devices via Bluetooth.

- **PIN:** parameterisable four-digit numeric value allowing secure connection and pairing with authorized user devices.

- **Track Information:** Enable the title of the track and the name of the artist. This functionality is exclusive of Bluetooth module.

- **Auxiliary Input:** enables music playback from a wired audio source, without Bluetooth connection.

  A choice can be made whether the auxiliary input has priority over Bluetooth streaming or vice versa. Thus, the playback control can only be carried out from the own priority user device audio player.

- **Playback Channels:** audio output channels. AudioInRoom has two independent stereo channels with two setup options for audio playback mode:

  ➢ **Stereo:** each of two tracks which form the audio signal is played through one of the speakers (or subchannels), recreating a more natural listening experience.

  ➢ **Mono:** the two tracks sound together through each speaker (or subchannel) and it is possible even disconnecting one of these speakers without altering the audio output.

- **Playback control** using KNX objects. There are two controls:

  ➢ **Play/Stop**: starts and pauses the audio playback of user device.

  ➢ **Skip**: plays the next track or plays back the previous one.

- **Volume Control.** AudioInRoom volume control depends directly on the Bluetooth of the device connected. Thus, three different volume controls are possible:

  ➢ **Device Volume Control**: audio player volume control of the user device.

  ➢ **AudioInRoom Volume Control**: volume control of amplifiers of AudioInRoom, by means of a communication object.

  ➢ **[Device + AudioInRoom]**:

    • Bidirectional behaviour: that makes that the signal volume of both amplifiers will be modified directly from the user device. Thus, when a volume command is sent through the communication object to the AudioInRoom, the user device audio player volume will be modified in the same way. iOS devices typically work with bidirectional communication.

    • Unidirectional behaviour: user device does not directly change the AudioInRoom volume. Simply it applies the percentage of its volume to the AudioInRoom volume, however both volumes are not synchronised. Android devices typically work with unidirectional communication.

> *Bidirectional behaviour:*
>
> - *AudioInRoom volume = 80%*
>
> - *User device volume changes to 20%*
>
> ➔ *[Device + AudioInRoom] Volume: = 20%*
>
> *Unidirectional behaviour:*
>
> - *AudioInRoom volume = 80%*
> - *User device volume changes to 20%*
> ➔ *[Device + AudioInRoom] Volume: = 16%*

Note: *In some Android devices, in advanced Bluetooth settings, there is a parameter called "Volume Synchronization" that, when activating, enables bidirectional communication.*

Regardless of the devices connected via Bluetooth, AudioInRoom can configure the following controls:

➢ **Initial Volume:** volume that will be set on each channel and/or subchannel after programming or a power loss. It is also the initial volume when the sound is played from the auxiliary input.

➢ **Volume Limitation:** maximum value for the output volume of each amplifier. The real output volume will be the result of adjusting the volume established by the user [0-100%] to the configured limit value, as follows:

> *[Limit] = 50%, [Control Volume] = 50% → [Output Volume] = 25%*
>
> *[Limit] = 60%, [Control Volume] = 60% → [Output Volume] = 36%*
>
> *[Limit] = 60%, [Control Volume] = 50% → [Output Volume] = 30%*

However, this limitation is transparent to the user. In other words, the value of the volume control object is shown on the bus in any case.

- **Mute audio output:** it is possible to mute the output volume in each channel and/or subchannel by communication object.

- **Tone:** AudioInRoom has, in addition to **Ring Tone**, a **Welcome** and **Wake Up Tone** ⚠. These three ringtones emit a beep through the speakers that will sound even when the Bluetooth module is off.

> ⚠ *For serial numbers prior to 20ACC0494 the welcome and wake up tones are not supported.*

- **Mute tones:** AudioInRoom can control whether or not the Ring Tone and the Welcome Tone sounds depending on the state of the room.

- **Alarms:** AudioInRoom provides three high temperature warning alarms, one for each channel and one for the device itself.

When an alarm occurs, audio playback is stopped. To deactivate the alarm, it is necessary to correct the problem that generates it (probably a connection error) and unfreeze it. The object for unfreezing the alarms is common to all of them.

ETS PARAMETERISATION

After enabling the Bluetooth audio module in the "General" screen (see section 2.1), a new tab will be incorporated into the tab tree on the left. The parameters available for configuring the Bluetooth module can be found in the "Configuration" subtab:

Figure 7. Audio Bluetooth Module Configuration.

- **On/Off Object** [*enabled/disabled*]: enables or disables the 1-bit object "**[Audio] On/Off**" which switch on / off the Bluetooth module.

  Moreover, the Bluetooth module status can be checked by reading the 1-bit object "**[Audio] On/Off (Status)**".

- **Device Name** [*Zennio AiR*]: default name of the Bluetooth module to identify the AudioInRoom when pairing. This is the name set when a Check-out process is performed.

  ➢ **Device Name Change Object** [*enabled/disabled*]: when enabled, it is possible to change the name of the device through the 14-bytes object "**[Audio][BT] Name**". The name saved in this object can be checked by reading the 14-bytes objects "**[Audio][BT] Name (Status)**". This is the name set when a Check-in process is performed.

    Note: *the status object "[Audio][BT] Name (status)" does not always indicate the current device name; it indicates the name that has been recorded in the object "[Audio][BT] Name" and that will be set on the device by Check-in.*

- **Pairing PIN Generation Mode**:

  ➢ [*Default*]: the PIN for the Bluetooth pairing is always 0000.

  ➢ [*Random*]: every time a check-in is made a random PIN is generated to be used during Bluetooth pairing, while with a check-out the PIN is restored to its default value.

  ➢ [*Custom*]: sets a PIN value by parameter, as well as the possibility of changing the PIN at runtime via the "**[Audio][BT] PIN**" object.

  ➢ **PIN** [*0000…9999*]: PIN value (default or custom, depending on the option selected above) of the Bluetooth module for device pairing. This is the PIN set when a Check-out process is performed.

  ➢ **PIN Object** [*enabled/disabled*]: enables the 14-bytes object "**[Audio][BT] PIN**" to modify the PIN value. The PIN saved in this object can be checked by reading the 14-bytes objects "**[Audio][BT] PIN (Status)**". This is the PIN set when a Check-in process is performed.

  Notes:

  - *If a PIN outside the allowed range [0000-9999] is entered in the "[Audio][BT] PIN" object:*

    o *If it is a numerical value, only the first 4 digits are taken.*

o *If it is not a numeric value, it is ignored and the PIN is updated to the value set by parameter.*

- *In check-in state if the PIN is changed via the object "**[Audio][BT] PIN**", this PIN will be used for pairing, while in check-out state, if the PIN is changed, it will be necessary to do a check-in for this change to take effect.*

- **Track Information:** [*enabled/disabled*]: enables o disables the objects "**[Audio] Title of the Track**" and "**[Audio] Name of the artist**"

- **Auxiliary Input** [*enabled/disabled*]: enables o disables, using the 1-bit object "**[Audio] Aux. Input**", the audio reception through the auxiliary input. The object "**[Audio] Aux. Input (Status)**" will indicate the status of that input.
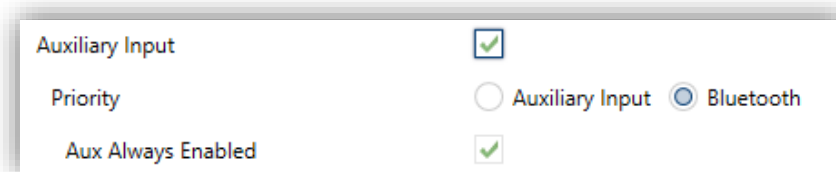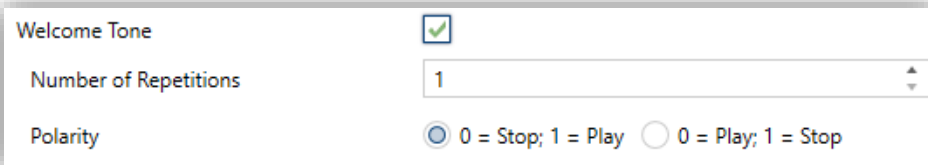
Figure 8. Auxiliary Input Priority.

➢ **Priority** [*Auxiliary Input / Bluetooth*]: sets which audio source, auxiliary input or Bluetooth streaming, will have the control over playback if both sources are active. So, when starting playback on the selected priority device, the audio of the secondary source will be stopped until the decoupling/disconnection of the first one is detected.

If the priority is assigned to the "Bluetooth" module, a new parameter will be displayed:

- **Aux Always Enabled [*enabled/disabled*].** If enabled, the auxiliary input will always be active. The binary objects **"[Audio] Aux. Input"** and **"[Audio] Aux. Input (Status)"** will not be needed anymore.

- **Absolute** [*enabled*]: volume control for each channel and/or subchannel through the 1-byte object "**[Audio] Channel X: Volume Absolute**" and/or "**[Audio] Channel X, Subchannel Y: Volume Absolute**", depending on whether the channel configuration is stereo or mono, respectively.

- **Relative Step** [*enabled/disabled*]: step volume control for each channel and/or subchannel through the 1-byte object "**[Audio] Channel X: Volume Relative Step**" and/or "**[Audio] Channel X, Subchannel Y: Volume Relative Step**", depending on whether the channel configuration is stereo or mono, respectively.

  - ➤ **Step Size** [*100% / 50% / 25% / 12.5% / 6.25% / 3.1% / 1.5%*]: value by which the volume of each channel and/or subchannel will increase or decrease when sending, respectively, "1" or "0" via the previous objects.

- **Relative** [*enabled/disabled*]: gradual volume control according to the step size received by the 4-bit object "**[Audio] Channel X: Volume Relative Control**" and/or "**[Audio] Channel X, Subchannel Y: Volume Relative Control**".

  - ➤ **Speed** [*3…10*] [*s*]: regulation speed to cover the entire control range (0% ↔ 100%).

The state object "**[Audio] Channel X: Volume (Status)**" and/or "**[Audio] Channel X, Subchannel Y: Volume (Status)**" will send the updated volume value each time a change through any of the above volume controls occurs.

- **Mute Polarity** *[0 = Disable Mute; 1 = Enable Mute / 0 = Enable Mute; 1 = Disable Mute*]: sets the polarity of the object to mute the volume.

- **Ring Tone** [*enabled*]: plays when a "0" or "1" is received through the 1-bit object "**[Audio] Ring Tone**"

- ⚠ **Welcome Tone** [*enabled/disabled*]: plays when a "1" is received through the 1-bit object "**[Audio] Welcome Tone**" and stops when a "0" is received.

Figure 9. Welcome Tone.

- ➤ **Number of Repetitions** [*1, 2*]: chooses the number of times the tone will be repeated.

➢ **Polarity** [*0 = Stop; 1 = Play / 0 = Play; 1 = Stop*]: sets the polarity of the object for the Welcome Tone.

⚠ **Wake Up Tone** [*enabled/disabled*]: plays when a "1" is received through the 1-bit object "**[Audio] Wake Up Tone**" and stops when a "0" is received.

Figure 10. Wake Up Tone.

➢ **Number of Repetitions** [*1…5…10*]: chooses the number of times the wake up tone will be repeated.

➢ **Polarity** [*0 = Stop; 1 = Play / 0 = Play; 1 = Stop*]: sets the polarity of the object.

⚠ *For serial numbers prior to 20ACC0494 the welcome and wake up tones are not supported.*

• **Volume** [*0…100*] [*%*]: volume of the ring, welcome and wake up tone.

• **Mute (Ring and Welcome Tone)** [*enabled/disabled*]: controls whether or not the sound of these tones will be muted depending on the room state.

Figure 11. Mute ring and welcome tone.

➢ **Type** [*1 bit / Room State*]: sets whether the ring and welcome tones will be muted via a 1-bit "**[Audio] Mute Tone**" or 1-byte "**[Audio] Room State**" object.

If the selected option is "1 bit", you can choose the **polarity** [*0 = Normal; 1 = Do Not Disturb / 0 = Do Not Disturb; 1 = Normal*] of the object.

If the selected option is "Room State", the tones will be muted when the room is in "Do not disturb" status.

- **Canal X** [*enabled/disabled*]: when enabled, the following parameters can be configured:

  ➢ **Configuration** [*Stereo / Mono*]: sets the audio playback mode.

  ➢ **Initial Volume** [*0…60…100*] [*%*]: percentage of volume after programming, power failure or after enabling the auxiliary input. If the selected payback mode is "Stereo", initial volume for **subchannel A** and **subchannel B** will be configured separately.

  ➢ **Volume Limitation** [*0…60…100*] [*%*]: maximum volume value.

    • **Object to Enable/Disable Volume Limitation** [*disabled/enabled*]: enables or disables the 1-bit control object "**[Audio] Channel X: Volume Limit**" and 1-bit status object "**[Audio] Channel X: Volume Limit (Status)**" for enabling or disabling the volume limitation and read the status.

    • **Volume Limitation Value Object** [*disabled/enabled*]: enables or disables the 1-byte control object "**[Audio] Channel X: Volume Limit Value**" and 1-byte status object "**[Audio] Channel X: Volume Limit Value (Status)**" for changing the volume limitation and read the status.

Furthermore, when a channel is enabled, the following objects are available:

➢ "**[Audio] Channel X: Mute**": silences each channel. If the "Mono" mode is set, the corresponding objects of each subchannel are also displayed: **[Audio] Channel X, Subchannel Y: Mute.** The mute control is independent of the volume control and takes precedence over it.

➢ "**[Audio] Channel X: Mute (Status)**" or "**[Audio] Channel X, subchannel Y: Mute (Status)**": objects to consult whether the channel or subchannel are muted, respectively.

> ➤ "**[Audio] Channel X: Alarm (Status)**": is sent with value' 1' to warn about a high temperature in the corresponding channel.

On the other hand, the following communication objects are always available:

- "**[Audio] Check-in/Check-out**": resets the Bluetooth module leaving it ready for a new user (Check-in) or returning to the default status when a user leaves (Check-out). See section 1.4.1 for details.

- "**[Audio] Check-in/Check-out (Status)**": will be sent at the end of each Check-in or Check-out procedure.

- "**[Audio][BT] Pairing**": enables (sending "<u>0</u>" or "<u>1</u>") the pairing mode and AudioInRoom remains visible via Bluetooth for approximately 180 seconds. See section 1.4.2 for details.

- "**[Audio][BT] Pairing Mode (Status)**": indicates the pairing process state. It is sent at the beginning and end of each pairing process

- "**[Audio][BT] Disconnect**": stops the connection of the devices to the AudioInRoom. The pairings remain.

- "**[Audio][BT] Play/Stop**": starts ('1') or pauses ('0') the audio playback. Only for Bluetooth connection.

- "**[Audio][BT] Play/Stop (Status)**": informs about the audio playback status. Only for Bluetooth connection.

- "**[Audio][BT] Skip**": plays the next ('1') or he previous ('0') track. Only for Bluetooth connection.

- "**[Audio] Alarm: Overheating (Status)**" is sent with value' 1' to warn about a high temperature in the device.

- "**[Audio] Unfreeze Alarms**": unfreezes the alarm.

- "**[Audio] Tone**"; 1-bit object that activates (sending "<u>0</u>" or "<u>1</u>") the ring tone.

ANNEX I. COMMUNICATION OBJECTS

- **"Functional range"** shows the values that, with independence of any other values permitted by the bus according to the object size, may be of any use or have a particular meaning because of the specifications or restrictions from both the KNX standard or the application programme itself.

Number	Size	I/O	Flags	Data type (DPT)	Functional Range	Name	Function
1	1 Bit		C - - T -	DPT_Trigger	0/1	[Heartbeat] Object to Send '1'	Sending of '1' Periodically
2	1 Bit	I	C - W - -	DPT_Switch	0/1	[Audio] On/Off	0 = Off; 1 = On
3	1 Bit	O	C R - T -	DPT_Switch	0/1	[Audio] On/Off (Status)	0 = Off; 1 = On
4	1 Bit	I	C - W - -	DPT_Switch	0/1	[Audio] Check-in/Check-out	0 = Check-out; 1 = Check-in
5	1 Bit	O	C R - T -	DPT_Switch	0/1	[Audio] Check-in/Check-out (Status)	0 = Check-out; 1 = Check-in
6	1 Bit	I	C - W - -	DPT_Trigger	0/1	[Audio][BT] Pairing	Launch Pairing Mode
7	1 Bit	O	C R - T -	DPT_State	0/1	[Audio][BT] Pairing Mode (Status)	0 = Pairing Mode Off; 1 = Pairing Mode On
8	1 Bit	I	C - W - -	DPT_Trigger	0/1	[Audio][BT] Disconnect	Disconnect All
9	14 Bytes	I	C - W - -	DPT_String_ASCII		[Audio][BT] Name	Device Name
10	14 Bytes	O	C R - T -	DPT_String_ASCII		[Audio][BT] Name (Status)	Device Name
11	14 Bytes	I	C - W - -	DPT_String_ASCII		[Audio][BT] PIN	PIN number (0000-9999)
12	14 Bytes	O	C R - T -	DPT_String_ASCII		[Audio][BT] PIN (Status)	PIN number
13	1 Bit	I	C - W - -	DPT_Enable	0/1	[Audio] Aux. Input	0 = Disable Aux. Input; 1 = Enable Aux. Input
14	1 Bit	O	C R - T -	DPT_Enable	0/1	[Audio] Aux. Input (Status)	0 = Aux. Input Disabled; 1 = Aux. Input Enabled
15	1 Bit	I	C - W - -	DPT_Start	0/1	[Audio][BT] Play/Stop	0 = Stop; 1 = Play
16	1 Bit	O	C R - T -	DPT_Start	0/1	[Audio][BT] Play/Stop (Status)	0 = Stop; 1 = Play
17	1 Bit	I	C - W - -	DPT_Step	0/1	[Audio][BT] Skip	0 = Backward; 1 = Forward
18, 40	1 Bit	I	C - W - -	DPT_Enable	0/1	[Audio] Channel X: Mute	0 = Disable Mute; 1 = Enable Mute
18, 40	1 Bit	I	C - W - -	DPT_Enable	0/1	[Audio] Channel X: Mute	0 = Enable Mute; 1 = Disable Mute
19, 41	1 Bit	O	C R - T -	DPT_Enable	0/1	[Audio] Channel X: Mute (Status)	0 = Mute Disabled; 1 = Mute Enabled
19, 41	1 Bit	O	C R - T -	DPT_Enable	0/1	[Audio] Channel X: Mute (Status)	0 = Mute Enabled; 1 = Mute Disabled
20, 22, 42, 44	1 Bit	I	C - W - -	DPT_Enable	0/1	[Audio] Channel X, Subchannel Y: Mute	0 = Disable Mute; 1 = Enable Mute
20, 22, 42, 44	1 Bit	I	C - W - -	DPT_Enable	0/1	[Audio] Channel X, Subchannel Y: Mute	0 = Enable Mute; 1 = Disable Mute
21, 23, 43, 45	1 Bit	O	C R - T -	DPT_Enable	0/1	[Audio] Channel X, Subchannel Y: Mute (Status)	0 = Mute Disabled; 1 = Mute Enabled
21, 23, 43, 45	1 Bit	O	C R - T -	DPT_Enable	0/1	[Audio] Channel X, Subchannel Y: Mute (Status)	0 = Mute Enabled; 1 = Mute Disabled

24, 46	1 Bit	I	C - W - -	DPT_Enable	0/1	[Audio] Channel X: Volume Limit	0 = Disable Volume Limit; 1 = Enable Volume Limit
25, 47	1 Bit	O	C R - T -	DPT_Enable	0/1	[Audio] Channel X: Volume Limit (Status)	0 = Volume Limit Disabled; 1 = Volume Limit Enabled
26, 48	1 Byte	I	C - W - -	DPT_Scaling	0% - 100%	[Audio] Channel X: Volume Limit Value	[0 … 100] %
27, 49	1 Byte	O	C R - T -	DPT_Scaling	0% - 100%	[Audio] Channel X: Volume Limit Value (Status)	[0 … 100] %
28, 50	1 Byte	I	C - W - -	DPT_Scaling	0% - 100%	[Audio] Channel X: Volume Absolute	[0 … 100] %
29, 51	1 Byte	O	C R - T -	DPT_Scaling	0% - 100%	[Audio] Channel X: Volume (Status)	[0 … 100] %
30, 32, 52, 54	1 Byte	I	C - W - -	DPT_Scaling	0% - 100%	[Audio] Channel X, Subchannel Y: Volume Absolute	[0 … 100] %
31, 33, 53, 55	1 Byte	O	C R - T -	DPT_Scaling	0% - 100%	[Audio] Channel X, Subchannel Y: Volume (Status)	[0 … 100] %
34, 56	1 Bit	I	C - W - -	DPT_Step	0/1	[Audio] Channel X: Volume Relative Step	0 = Decrease; 1 = Increase
35, 36, 57, 58	1 Bit	I	C - W - -	DPT_Step	0/1	[Audio] Channel X, Subchannel Y: Volume Relative Step	0 = Decrease; 1 = Increase
37, 59	4 Bit	I	C - W - -	DPT_Control_Dimming	0x0 (Stop) 0x1 (Dec. by 100%) … 0x7 (Dec. by 1%) 0x8 (Stop) 0xD (Inc. by 100%) … 0xF (Inc. by 1%)	[Audio] Channel X: Volume Relative Control	4 Bits Relative Control
38, 39, 60, 61	4 Bit	I	C - W - -	DPT_Control_Dimming	0x0 (Stop) 0x1 (Dec. by 100%) … 0x7 (Dec. by 1%) 0x8 (Stop) 0xD (Inc. by 100%) … 0xF (Inc. by 1%)	[Audio] Channel X, Subchannel Y: Volume Relative Control	4 Bits Relative Control
62	1 Bit	I	C - W - -	DPT_Ack	0/1	[Audio] Unfreeze Alarms	[Alarm=Inactive] + [Unfreeze=1] => End Alarm
63, 64	1 Bit	O	C R - T -	DPT_Alarm	0/1	[Audio] Channel X: Alarm (Status)	0 = No Alarm; 1 = Alarm
65	1 Bit	O	C R - T -	DPT_Alarm	0/1	[Audio] Alarm: Overheating (Status)	0 = No Alarm; 1 = Alarm
66	1 Bit	I	C - W - -	DPT_Trigger	0/1	[Audio] Ring Tone	Ring Tone
67 ⚠	1 Bit	I	C - W - -	DPT_Start	0/1	[Audio] Welcome Tone	0 = Play; 1 = Stop
	1 Bit	I	C - W - -	DPT_Start	0/1	[Audio] Welcome Tone	0 = Stop; 1 = Play
68 ⚠	1 Bit	I	C - W - -	DPT_Start	0/1	[Audio] Wake Up Tone	0 = Play; 1 = Stop
	1 Bit	I	C - W - -	DPT_Start	0/1	[Audio] Wake Up Tone	0 = Stop; 1 = Play
69	1 Byte	I	C - W - -	DPT_Room_State	0/1	[Audio] Room State	0 = Normal; 1 = Make-Up Room; 2 = Do Not Disturb

	1 Bit	I	C - W - -	DPT_Enable	0/1	[Audio] Mute Tone	0 = Normal; 1 = Do Not Disturb
	1 Bit	I	C - W - -	DPT_Enable	0/1	[Audio] Mute Tone	0 = Do Not Disturb; 1 = Normal
70	14 Bytes	O	C R - T -	DPT_String_ASCII		[Audio] Title of the Track	Title of the Track
71	14 Bytes	O	C R - T -	DPT_String_ASCII		[Audio] Name of the Artist	Name of the Artist

⚠️ *For serial numbers prior to 20ACC0494 the welcome and wake up tones are not supported.*

Join and send us your inquiries
about Zennio devices:
https://support.zennio.com