ModusToolbox™ Audio Front-End Configurator user guide

DEEPCRAFT™ Audio Enhancement Tech Pack

ModusToolbox™ DEEPCRAFT™ Audio Enhancement Tech Pack version 1.1.0

Audio Front-End Configurator version 1.60.0

About this document

Scope and purpose

The Audio Front-End Configurator is a stand-alone tool included with the ModusToolbox™ software used to generate configuration for the Audio Front-End middleware. In addition, the tool offers real-time tuning capabilities, enabling the user to dynamically adjust parameters directly on the target device, ensuring optimal audio performance.

Intended audience

This document helps application developers understand how to use the Audio Front-End Configurator as part of creating a ModusToolbox™ application.

Document conventions

Convention	Explanation
Bold	Emphasizes heading levels, column headings, menus and sub-menus
Italics	Denotes file names and paths.
`Courier New`	Denotes APIs, functions, interrupt handlers, events, data types, error handlers, file/folder names, directories, command line inputs, code snippets
File > New	Indicates that a cascading sub-menu opens when you select a menu item

Abbreviations and definitions

The following define the abbreviations and terms used in this document:

- AFE – audio front end
- AEC – acoustic echo cancellation
- Configurator – A GUI-based tool used to configure a resource.

Reference documents

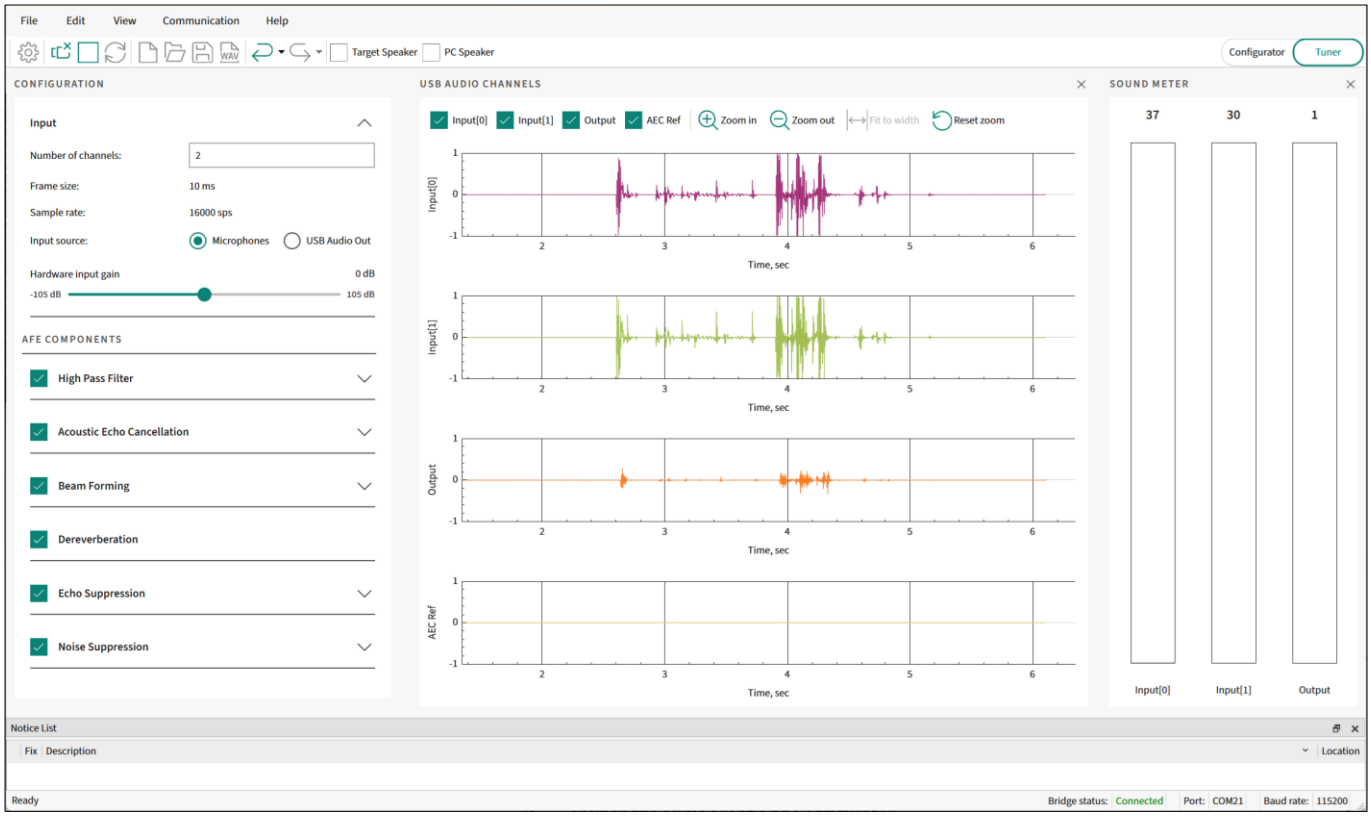
Refer to the following documents for more information as needed:

- ModusToolbox™ tools package user guide
- Eclipse IDE for ModusToolbox™ user guide
- VS Code for ModusToolbox™ user guide
- Device Configurator user guide
- Device technical reference manuals

Table of contents

1 Overview

The Audio Front-End Configurator provides a graphical user interface to create and edit Audio Front-End peripheral configuration.



2 Launch the Audio Front-End Configurator

There are numerous ways to launch the Audio Front-End Configurator, and those ways depend on how you use the various tools in ModusToolbox™ software.

2.1 make command

As described in the ModusToolbox™ tools package user guide "ModusToolbox™ build system" chapter, you can run numerous make commands in the application directory, such as launching the Audio Front-End Configurator. After you have created a ModusToolbox™ application, navigate to the application directory and type the following command in the appropriate bash terminal window:

```
make audio-fe-configurator
```

This command opens the Audio Front-End Configurator GUI for the specific application in which you are working.

2.2 VS Code and Eclipse

VS Code and Eclipse have tools to launch the Audio Front-End Configurator from within an open application. Refer to the applicable user guide for more details:

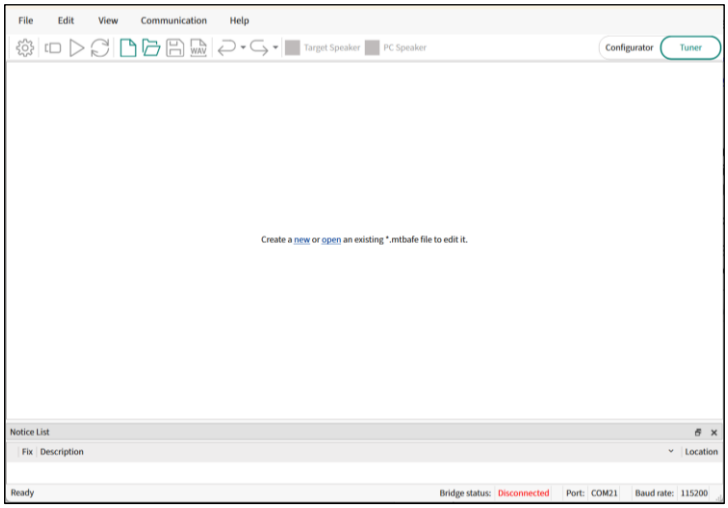
- Eclipse IDE for ModusToolbox™ user guide
- VS Code for ModusToolbox™ user guide

2.3 Executable (GUI)

If you don't have an application or if you just want to see what the configurator looks like, you can launch the Audio Front-End Configurator GUI by running its executable as appropriate for your operating system (for example, double-click it or select it using the Windows **Start** menu). By default, it is installed here:

<install_dir>/ModusToolbox/packs/DEEPCRAFT-Audio-Enhancement-Tech-Pack/tools/audio-fe-configurator

When launched this way, the Audio Front-End Configurator opens without any settings configured. You can either open a specific configuration file or create a new one. See Menus for more information.



2.4 Executable (CLI)

The Audio Front-End Configurator executable can be run from the command line, and it also has a "cli" version of the executable as well. Running configurator executables from the command line can be useful as part of batch files or shell scripts to re-generate the source code based on the latest configuration settings. The exit code for the executable is zero if the operation is successful, or non-zero if the operation encounters an error. For more information about the command-line options, run the executable using the `-h` option.

2.5 Dependencies

The tool is supported on Windows, Linux, and macOS.

GStreamer

Under Linux, the Audio Front-End Configurator uses Qt library *libQt6Multimedia.so.6.8.*3, which calls the functions from the *GStreamer* libraries. Before starting the Audio Front-End Configurator, install the following shared libraries from the *GStreamer* package, version 1.0.so.0:

- libgstpbutils
- libgstaudio
- libgstapp
- libgstallocators
- libgstgl
- libgstvideo
- libgstbase
- libgstreamer

FFmpeg

The Configurator uses the Qt 6.8.3 Multimedia module, which internally utilizes the FFmpeg 7.1 library. FFmpeg is distributed under the LGPLv2.1 or later versions license. Although FFmpeg includes optional components licensed under GPL or LGPLv3, these components are not included in the binaries provided with Qt.

3 Quick start

This section provides a simple workflow for how to use the Audio Front-End Configurator.

1. The Configurator requires the Audio Front-End Configurator configuration file provided in a corresponding code example.
2. Launch the Audio Front-End Configurator and open the configuration file from the code example.
3. Select the **Number of channels** to define either mono or stereo mode.
4. Select **Microphones** as the input source.
5. Disable **Noise Suppression**.
6. Save the configuration. See Code generation.
7. Build the project and flash it to the device.
8. Configure the connection parameters in the **Connection Settings**.
9. Connect the Configurator to the device.
10. Run the system.
11. The obtained audio data will be displayed on the plots.

4 Code generation

The Configurator generates header (*cy_afe_configurator_settings.h*) and source (*cy_afe_configurator_settings.c*) files that contain relevant firmware used by the Audio Front-End middleware configuration and operation. The generated *.h* and *.c* files are located in the *"GeneratedSource"* folder next to the **. mtbafe* file, which contains the user configuration.

Note: The Audio Front-End Configurator does not interact with the design.modus file.

5 GUI description

The Audio Front-End Configurator GUI contains menus to configure the audio front-end settings and a Notice List to provide indications.

5.1 Menus

5.1.1 File

- **New** – Creates a new file with new configuration.
- **Open…** – Opens the configuration file (*.mtbafe*).
- **Open in System Explorer** – Opens your computer's file explorer tool to the folder that contains the **.mtbafe* file.
- **Close** – Closes the *.mtbafe* configuration file.
- **Save** – Saves the existing configuration into *.mtbafe* file and generates C files.
- **Save As…** – Saves the existing file under a different name.
- **Import…** – Imports a specified configuration file.
- **Export…** – Exports the current configuration file into a specified file.
- **Load WAV Files**… – Opens a dialog window to load WAV files for the following options: Target Speaker or Audio Input, PC Speaker.
- **Save Recording to File…** – Saves the recorded audio data from the device to a file.
- Recent files – Displays recent files that you can open directly.
- **Exit** – Closes the configurator.

5.1.2 Edit

- **Undo** – Undoes the last action or sequence of actions.
- **Redo** – Redoes the last undone action or sequence of undone actions.

5.1.3 View

- **Notice List** – Displays/hides Notice List. See the Device Configurator user guide for more information about the Notice List.
- **Toolbar** – Displays/hides the toolbar.
- **USB Audio Channels** – Displays/hides this.
- **Sound Meter** – Displays/hides this.
- **Configurator Mode** – Switches the GUI layout to the configuration parameters only without the possibility to connect to the device.
- **Tuner Mode** – Switches the GUI layout to enable the tuning functionality.
- **Reset View** – Resets the view to the default.

5.1.4 Communication

- **Connect/Disconnect Device** – Connects to/Disconnects from the device via the selected communication channel.
- **Run/Stop System** – Runs the system to stream audio and collect information from the target device.

- **Sync Parameters...** – Opens a window to load filter settings from the Configurator to the device and vice versa.
- **Target Speaker** – Enables/disables a speaker controlled by the target device. The Configurator can stream audio data over the USB to play on this speaker. The audio source comes from the **Target speaker** or **Audio input** In the **Load WAV Files** dialog.
- **PC Speaker** – Enables/disables a PC-controlled speaker. The Configurator can play directly to this speaker. The audio source comes from the **PC Speaker** WAV file in **Load WAV Files** dialog.
- **Connection Settings ...** – Opens a window to configure the connection parameters.

5.1.5 Help

- **View Help** – Opens this document.
- **About Audio Front-End Configurator** – Opens the About box to display the version information, with links to open https://www.infineon.com and the current session log file.

5.2 Toolbar

Use the check box under the View menu to display or hide the toolbar.

The toolbar contains common commands from the File, Edit, View, and Communication menus in the respective order: **Connection Settings**, **Connect**/**Disconnect device**, **Run/Stop system**, **Sync Parameters**, **New**, **Open**, **Save, Save Recording to File**, **Undo/Redo**, **Target Speaker**, **PC Speaker, Configurator/Tuner** mode.

5.3 Status bar

The Status bar on the left side displays the various status messages and input field ranges, etc. On the right side, it displays the communication parameters.

5.4 Notice List

The Notice List pane combines notices (errors, warnings, tasks, and notes) from many places in the configuration into a centralized list. If a notice shows a location, you can double-click that entry to display the parameter causing the error or warning.

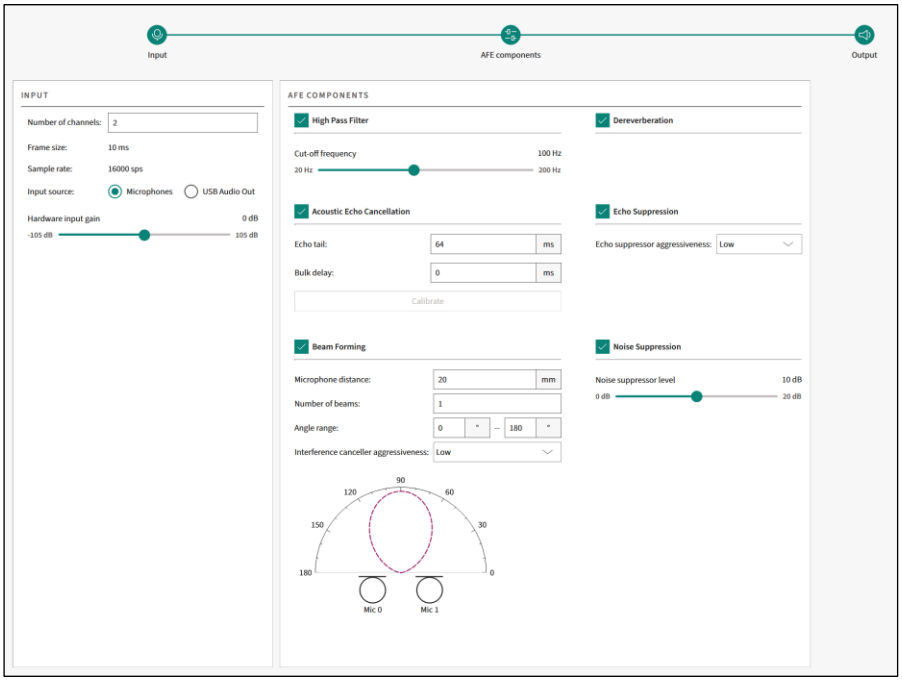
For more information, refer to the Device Configurator user guide.

6 Parameters Configuration

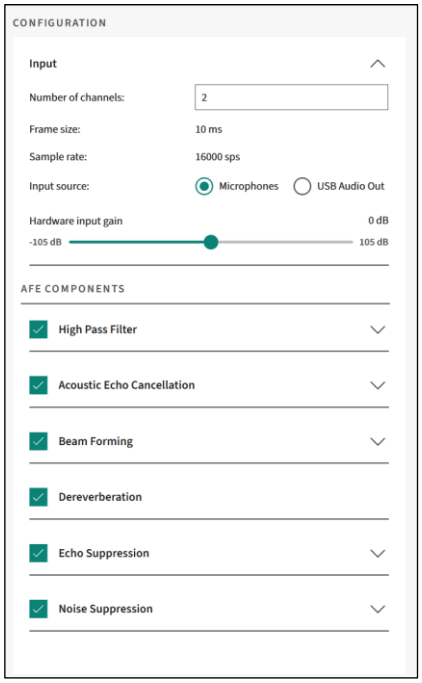
The main window displays the configuration pane dedicated for editing audio front-end profile settings necessary for the audio front-end firmware to run the audio processing chain. These parameters values are stored in the *.mtbafe* file and translated into the *.c* and *.h* files.

Depending on the GUI mode, it looks as follows.

Configurator view

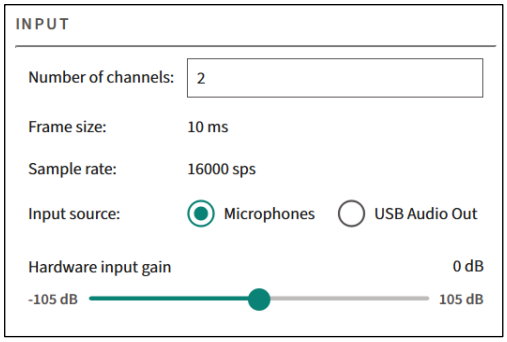


Tuner view



6.1 Input

The **Input** pane allows you to define the key parameters relating to the audio input stage.

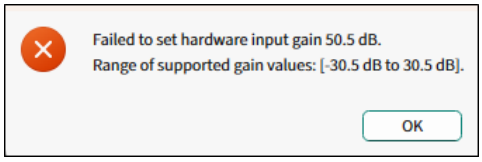


On the **Input** pane, you can configure the following:

- **Number of channels** – Specifies the number of input audio channels to be used. Range: 1 to 2 channels.
- **Frame size (ms)** – Defines the fixed duration of each audio frame. Set to 10 ms (not configurable).
- **Sample rate (sps)** – Specifies the audio signal's fixed sampling rate. Set to 16000 sps (not configurable).
- **Input source** – Defines the origin of the incoming audio signal. Options: **Microphones** or **USB Audio Out**.

- **Hardware input gain** – Adjusts the hardware-level input gain for the incoming audio signal. Range: -105 dB to 105 dB.

  Note: If the device was programmed with values from an input gain range different from the default range in the Configurator, and then you set some input gain value from the default range, you may see the following message:

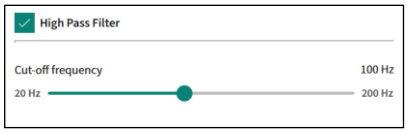


6.2 AFE Components

The **AFE Components** section provides configuration of the audio processing algorithms, such as **High Pass Filter**, **Acoustic Echo Cancellation**, **Beam Forming**, **Dereverberation**, **Echo Suppression**, **Noise Suppression.**

6.2.1 High Pass Filter

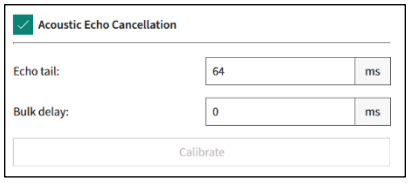
A High Pass Filter is designed to allow frequencies above a certain threshold, the **Cut-off frequency**, to pass through while attenuating lower frequencies.



On the **High Pass Filter** pane, you can configure the **Cut-off frequency** parameter. It defines the threshold in Hz at which the High Pass Filter begins attenuating lower frequencies. Range: from 20 to 200 Hz.

6.2.2 Acoustic Echo Cancellation

Acoustic Echo Cancellation is an advanced signal processing technique used to eliminate echoes that occur during audio communication. Typically, they are caused by the feedback of the audio output (e.g., from speakers) being picked up by the input device (e.g., a microphone).



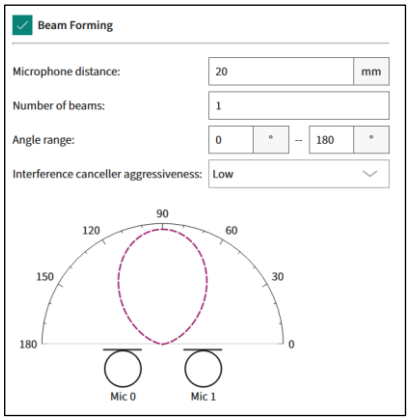
On the **Acoustic Echo Cancellation** pane, you can configure the following:

- **Echo tail** – Configured relatively to the time window, which the AEC algorithm analyzes in determining echo cancellation. The range is from 0 to 64 ms.
- **Bulk delay** – Configured relatively to the delay due to the audio buffering to the speaker. The range is from 0 to 64 ms.
- **Calibrate** – Click this button to automatically calibrate the bulk delay. It is enabled when the device is connected.

6.2.3 Beam Forming

Beam Forming is a signal processing technique used to control the directionality of the sound capture or transmission. It works by combining two microphone signals to focus on the sound from a specific direction while suppressing noise or interference from the other directions.

You can configure the **Beam Forming** only if two channels are used, which implies that internal calibration parameters of the input microphones are set. The pane displays the actual beam forming on a graph based on given configuration.



On the **Beam forming parameters** pane, you can configure the following:

- **Microphones distance** – Distance between the microphones. Range: 4 to 100 mm.
- **Number of beams** – Specifies the number of directional audio paths created by the beam forming algorithm. Range: 2 to 12 beams.
- **Angle range** – Defines the angular coverage of the beam forming system, measured in degrees. Range: 0° to 180°.
- **Interference canceller aggressiveness** – Controls the intensity of the interference cancellation. Options: Low, Medium, High.

6.2.4 Dereverberation

Dereverberation is a signal processing technique used to reduce or eliminate the effects of reverberation in audio signals. Reverberation occurs when sound waves reflect off surfaces in an enclosed environment, creating a prolonged and often undesired echo-like effect that can degrade audio clarity.

The **Dereverberation** does not have additional configuration parameters.

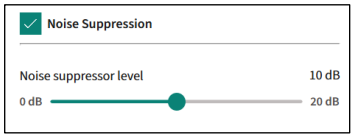
6.2.5 Echo Suppression

Echo suppression is a signal processing technique designed to mitigate or eliminate echo artifacts in audio systems. Echo suppression reduces the strength of the reflected or looped audio signal by attenuating it before it becomes perceptible. Unlike **Acoustic Echo Cancellation (AEC)**, which predicts and dynamically removes the echo signal through advanced filtering, echo suppression simply detects and applies gain reduction or attenuation to signals considered to be echo. This is often less complex than cancellation but may have limitations in challenging acoustic scenarios.

- **Echo suppressor aggressiveness** –Defines the degree of attenuation applied to echoes in the audio signal. Options: Low, Medium, High; Default: Low.

6.2.6 Noise Suppression

Noise suppression is a technique used to reduce or remove unwanted background noise from an audio signal.



- **Noise suppressor level** – Determines the degree to which background noise is reduced during audio processing. Higher values correspond to more aggressive noise suppression, effectively removing louder background noise. Range: 0-20 dB; Default: 10 dB.

7 Tuner

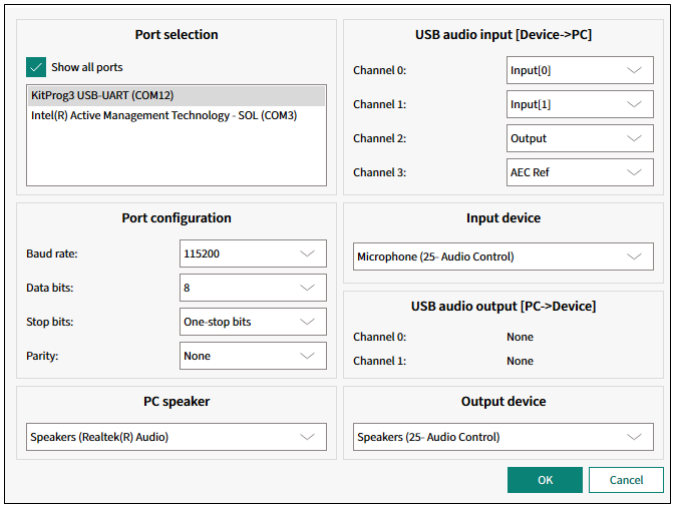
The AFE Configurator provides tuning capabilities designed to facilitate real-time audio configuration and analysis. It allows the user to connect directly to the target device and provides visual feedback, enabling tuning of audio parameters. To enable tuning capabilities, switch to the **Tuner** GUI mode.

Key features:

- **Device Connectivity**
  The GUI supports connection to the target device through enabling live monitoring its audio processing performance.
- **Dynamic Parameter Tuning**
  You can modify configuration parameters dynamically and apply changes to the connected device on the fly, without needing to restart or redeploy. This capability supports iterative tuning for optimal performance in real-time environments.
- **USB Audio Channel Plots**
  The interface displays dynamic visual plots of USB audio channels providing a real-time representation of audio signals. This helps the user analyze the signal behavior and identify potential issues or optimization opportunities.
- **Sound Meter**
  A built-in sound level meter provides accurate measurements of audio intensity. This visual aid ensures proper signal levels for your applications.

7.1 Connection Settings

This command opens a dialog to configure connection settings.



7.1.1 Port selection

The Port selection list displays available communication ports that can be used for UART interface connections. By default, only KitProg and MiniProg devices are shown there.

- **Show all ports** – Check this if you need a custom UART adapter (not the recommended KitProg and MiniProg devices).

7.1.2 Port configuration

The Port Configuration settings define the communication parameters for the UART interface, allowing proper data transmission and reception. The configuration options are as follows:

- **Baud rate** – Specifies the speed of data transmission. Range: 1200-115200 bps.
- **Data bits** – Defines the number of bits used to represent the data in each character during communication. Options: 7; 8.
- **Stop bits** – Indicate the end of a data packet during communication. Options: One-stop bits; One-and-half-stop bits; Two-stop bits.
- **Parity** – Adds a form of error-checking to the data transmission by appending a parity bit. Options: None; Odd; Even.

7.1.3 PC speaker

PC speaker is a speaker controlled by the PC that runs this tool. The Configurator can play directly to this speaker. The audio source comes from the PC speaker WAV file selected in the **Load WAV Files** dialog.

7.1.4 USB audio input [Device->PC]

The USB audio input has four channels, which can be linked to the following signals**:**

- **Input[0]** – The first input channel.
- **Input[1]** – The second input channel (stereo mode only).
- **Output** – Audio front-end output signal (post-processed).
- **AEC Ref** – AEC reference signal (also played to the Target speaker).
- **AEC_Out_Ch0** – AEC output, the first channel.
- **AEC_Out_Ch1** – AEC output, the second channel.
- **BF_Out** – Beam-forming output.
- **DR_Out** – Dereverberation output.

7.1.5 Input device

Specifies the audio source for incoming signals to the system. Select the required from the drop-down list.

*Note: The **Input** and **Output** must be selected for the same device.*

7.1.6 USB audio output [PC->Device]

USB audio output supports two channels, which might be used to play sound in the target speaker or to overwrite the audio input. Channels 0 and Channel 1 usage depends on the selected parameters: **Input source** and **Target speaker**.

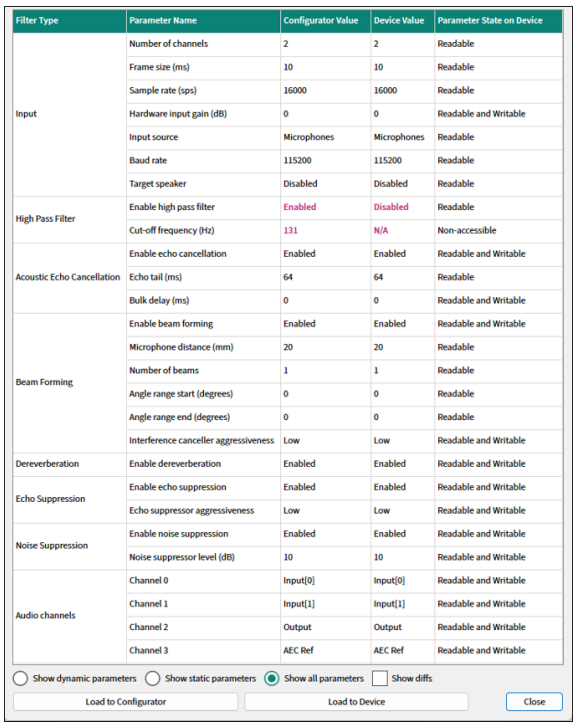
7.1.7 Output device

Specifies where the processed audio will be routed for playback. Select the required from the drop-down list.

*Note: The **Input** and **Output** must be selected for the same device.*

7.2 Sync Parameters

Click the **Sync Parameters** command to see the "Sync Parameters" dialog. This displays the difference between Configurator settings and settings programmed to Device.



Filter Type	Parameter Name	Configurator Value	Device Value	Parameter State on Device
Input	Number of channels	2	2	Readable
	Frame size (ms)	10	10	Readable
	Sample rate (sps)	16000	16000	Readable
	Hardware input gain (dB)	0	0	Readable and Writable
	Input source	Microphones	Microphones	Readable
	Baud rate	115200	115200	Readable
	Target speaker	Disabled	Disabled	Readable
High Pass Filter	Enable high pass filter	Enabled	Disabled	Readable
	Cut-off frequency (Hz)	131	N/A	Non-accessible
Acoustic Echo Cancellation	Enable echo cancellation	Enabled	Enabled	Readable and Writable
	Echo tail (ms)	64	64	Readable
	Bulk delay (ms)	0	0	Readable and Writable
Beam Forming	Enable beam forming	Enabled	Enabled	Readable and Writable
	Microphone distance (mm)	20	20	Readable
	Number of beams	1	1	Readable
	Angle range start (degrees)	0	0	Readable
	Angle range end (degrees)	0	0	Readable
	Interference canceller aggressiveness	Low	Low	Readable and Writable
Dereverberation	Enable dereverberation	Enabled	Enabled	Readable and Writable
Echo Suppression	Enable echo suppression	Enabled	Enabled	Readable and Writable
	Echo suppressor aggressiveness	Low	Low	Readable and Writable
Noise Suppression	Enable noise suppression	Enabled	Enabled	Readable and Writable
	Noise suppressor level (dB)	10	10	Readable and Writable
Audio channels	Channel 0	Input[0]	Input[0]	Readable and Writable
	Channel 1	Input[1]	Input[1]	Readable and Writable
	Channel 2	Output	Output	Readable and Writable
	Channel 3	AEC Ref	AEC Ref	Readable and Writable

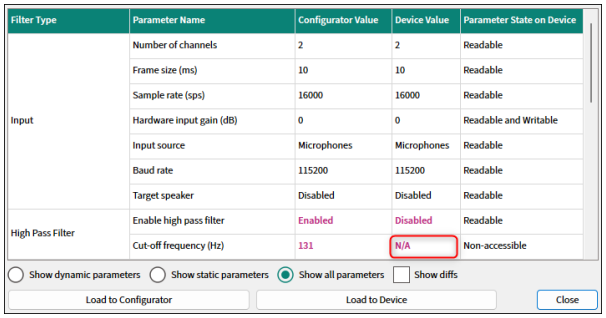
Show dynamic parameters | Show static parameters | Show all parameters | Show diffs

Load to Configurator | Load to Device | Close

Use the **Show dynamic parameters**, **Show static parameters**, **Show all parameters**, **Show diffs** buttons to display and compare different parameters.

To transfer the filter settings defined in the device to the Audio Front-End Configurator, click the **Load to Configurator** button. To transfer the filter settings defined in the Configurator, click the **Load to Device** button.

Some filters parameters (if the filter was flashed in the disabled state) cannot be readable. So, these parameters cannot be updated during transfer operation. Their values are marked by "N/A".



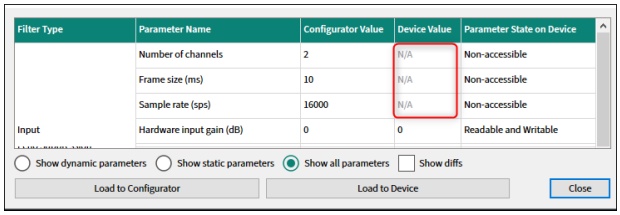
Filter Type	Parameter Name	Configurator Value	Device Value	Parameter State on Device
Input	Number of channels	2	2	Readable
	Frame size (ms)	10	10	Readable
	Sample rate (sps)	16000	16000	Readable
	Hardware input gain (dB)	0	0	Readable and Writable
	Input source	Microphones	Microphones	Readable
	Baud rate	115200	115200	Readable
	Target speaker	Disabled	Disabled	Readable
High Pass Filter	Enable high pass filter	Enabled	Disabled	Readable
	Cut-off frequency (Hz)	131	N/A	Non-accessible

Show dynamic parameters | Show static parameters | Show all parameters | Show diffs

Load to Configurator | Load to Device | Close

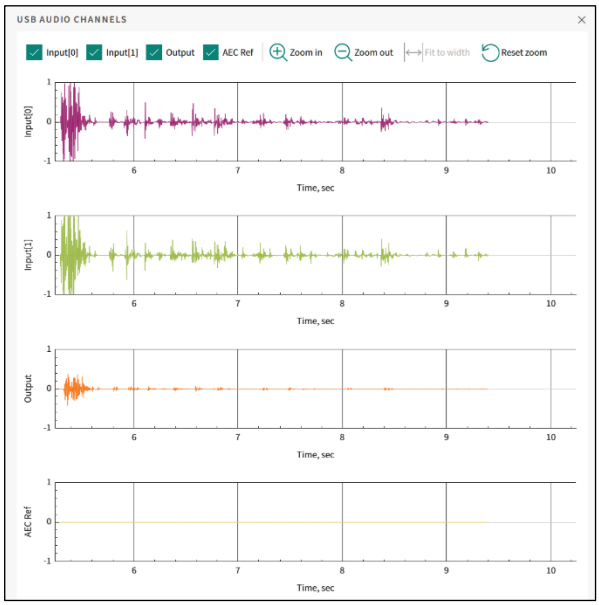
To update them: save the configuration, rebuild the project, and program the device.

If the project does not support the static parameters reading, their values are marked by "N/A" grayed out.



7.3 USB Audio Channels

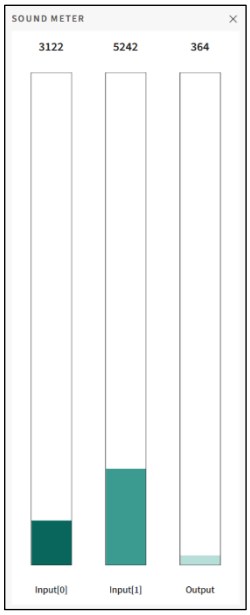
The two tabs **USB Audio Channels** and **Sound Meter** are sets of graphs that allow assessing given configuration. These tabs can be split into two separate views and docked anywhere on the Configurator window.



- Graphs
  - Check the **Input(0)**, **Input(1)**, **Output**, and **AEC Ref** channels checkboxes to display corresponding graphs. You can change the channels in the "Connection Settings" dialog.
- Zoom settings
  - **Zoom in**, **Zoom out** – Use these buttons to examine graphs in more details. Also, you can use the mouse wheel.
  - **Fit to width** – Click this to display graphs in the full view.
  - **Reset zoom** – Removes zoom, displays graphs in the 1:1 proportion.

7.4 Sound Meter

For stereo mode, visualizes the channels' levels **Input (0)**, **Input (1)**, and **Output**. For mono mode, visualizes the channels' levels **Input (0)** and **Output**.



8 Known issues

Problem	Workaround

The Audio Front-End Configurator does not receive audio signals from a device on Ubuntu 24.04.	WARNING: This may break your Ubuntu installation!
	Switch to the pulseaudio audio backend compatible with QtMultimedia from Qt 6.2.8.
	`sudo apt install pulseaudio pulseaudio-module-bluetooth gstreamer1.0-pulseaudio`
	`systemctl --user stop pipewire.socket pipewire-pulse.socket`
	`systemctl --user disable pipewire.socket pipewire-pulse.socket`
	`systemctl --user mask pipewire.socket pipewire-pulse.socket`
	`systemctl --user enable pulseaudio.service pulseaudio.socket`
	`systemctl --user start pulseaudio.service pulseaudio.socket`

9 Version changes

This section lists and describes the changes for each version of this Configurator.

Version	Change descriptions
1.0	New tool.
1.10	PSOC™ Edge 84 Early Access Pack 0.2.4
1.20	PSOC™ Edge 84 Early Access Pack 0.2.5
1.40	PSOC™ Edge E84 Early Access Pack 0.3.3 • Added support of UART commands to read static parameters values • Added informative messages about Configurator and device parameter mismatch • Improved audio data visualization
1.50	• Switched to Qt 6.8.3, which uses FFmpeg library backend. • Changed the GUI layout. Two modes are available: configurator and tuner. • Fixed issues with disconnecting the device in runtime.
1.60	Added support of input gain range dynamic reading.

Revision history

Revision	Date	Description
**	2024-05-20	New document. New document. PSOC™ Edge 84 Early Access Pack 0.2.4.
*A	2024-05-24	Added section "Dependencies".
*B	2024-10-22	Updated to version 1.20.
*C	2025-04-29	Updated to version 1.40.
*D	2025-09-12	Updated to version 1.50.
*E	2025-10-07	Updated to version 1.60.

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.

Edition 2025-10-07

Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about this document?

Email: erratum@infineon.com

Document reference

002-39990 Rev. *E

Important notice
The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics (“Beschaffenheitsgarantie”)

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer’s compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer’s products and any use of the product of Infineon Technologies in customer’s applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer’s technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

Warnings
Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies’ products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.