Product Document

Published by
ams OSRAM Group

User Guide

UG000480

AS3460

Digital Augmented Hearing Demo Kit

AS3460_EVALUATION_BOARD

v4-00 • 2022-Jan-05

Content Guide

1 Introduction

The AS3460 is a digital augmented hearing companion device ideally suited for stereo headsets and headphones as well as True Wireless applications. The extremely small form factor enables unique augmented hearing experience for its end user with negligible influence on overall playtime of the headset due to ultra-low power consumption. The high integration factor of AS3460 combined with its all new Augmented Hearing Engine do allow for lowest power signal processing especially tailored for the needs of active noise cancelling- and augmented hearing applications. The powerful low latency digital signal processor (AHE) supports up to 250 Bi-Quad calculations to ideally match the requirements of todays sophisticated acoustics designs and small form factors. Unique digital signal processor (DSP) functions like seamless ANC crossfading or its integrated music feedback compensation engine do allow lifting up user’s augmented hearing experience to its next level.

The development phase of such applications could be challenging and with this document, there are some guideline listed to support customers in the product design. Below is a detailed description of the AS3460 Evaluation Board and the configuration software FleX. Also some instructions are given regarding ANC filter design.

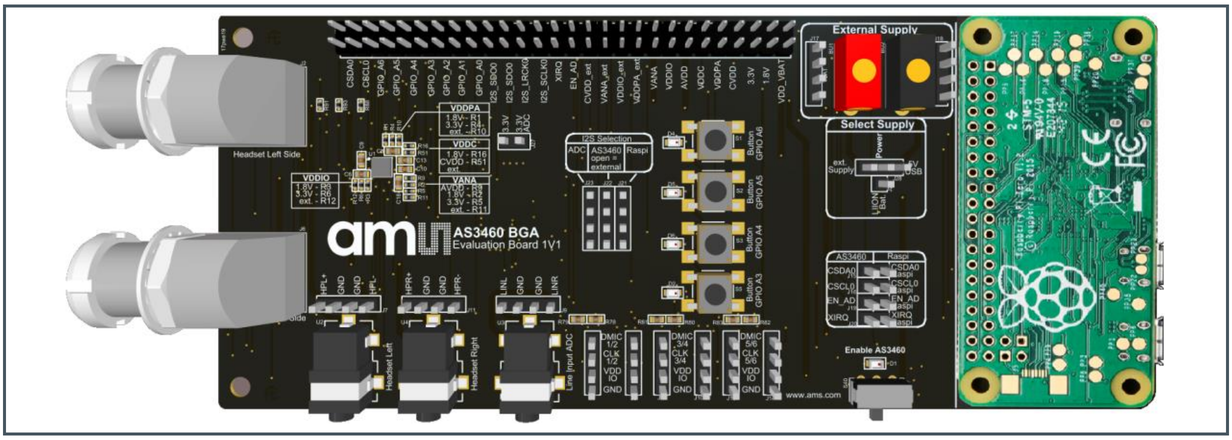
The first section contains a quick start guide for the Evaluation Kit.

1.1 Evaluation Kit

The AS3460 Evaluation Kit package includes the following:

- AS3460 Evaluation Board

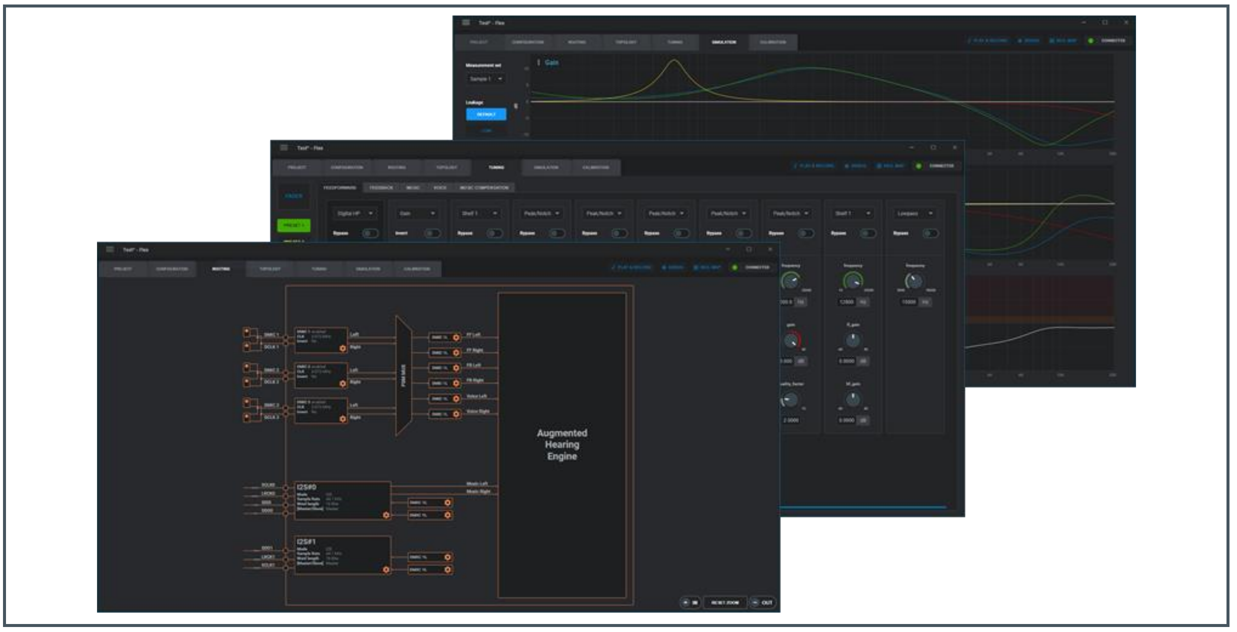
Figure 1:
Evaluation Board Hardware



- FleX Filter Design Tool

This tool allows customers to design filters for the AS3460 ANC device.

Figure 2:
FleX Filter Design Tool Software



1.2 Ordering Information

Ordering Code	Description
AS3460_EVALUATION_BOARD	AS3460 Digital Augmented Hearing Demo Kit

2 Quick Start Guide

This section will explain some key features of FleX required for starting an ANC-Filter project.

To install the latest FleX software please visit: ams.com and login with your email address and your password. If you do not have a login yet, please request one by clicking on “Create Account“ in the “Login” section.

When successfully logged in, you can download the latest evaluation software in the products section under tools.

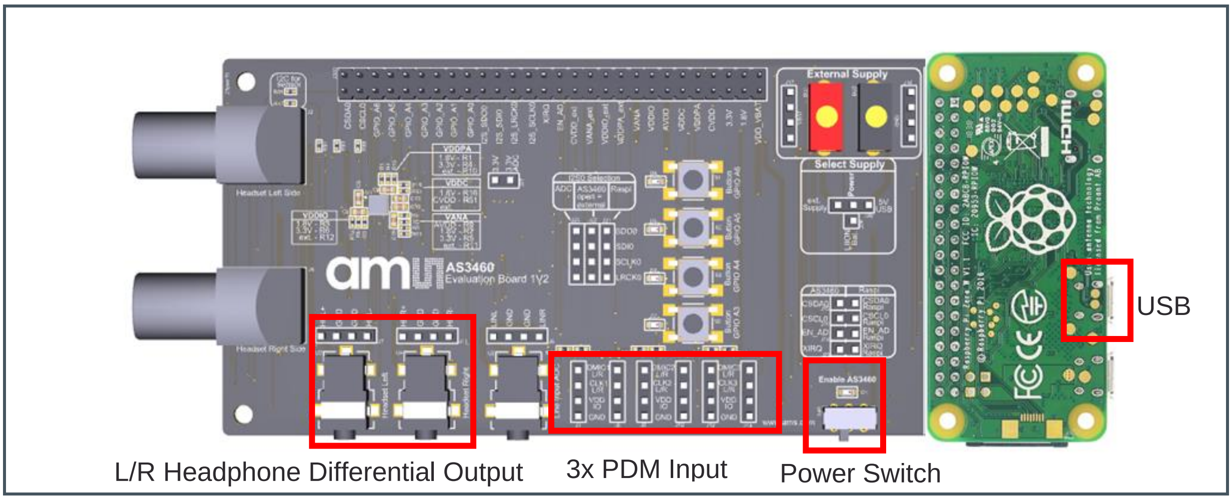
Please make sure that you have connected the AS3460 Evaluation Board to your PC before you install FleX Software the first time. This is mandatory for proper device driver installation.

2.1 Hardware Setup

To use FleX, it is necessary to connect the AS3460 Evaluation Board’s RaspberryPi via USB to a windows computer. When connecting the RaspberryPi for the first time, it will show up as a mass storage device, which contains the drivers that are required for FleX to function properly.

When creating a new project or opening an existing one, FleX will connect to the evaluation board. The default IP-address and port for this is **169.254.0.2:8080.**

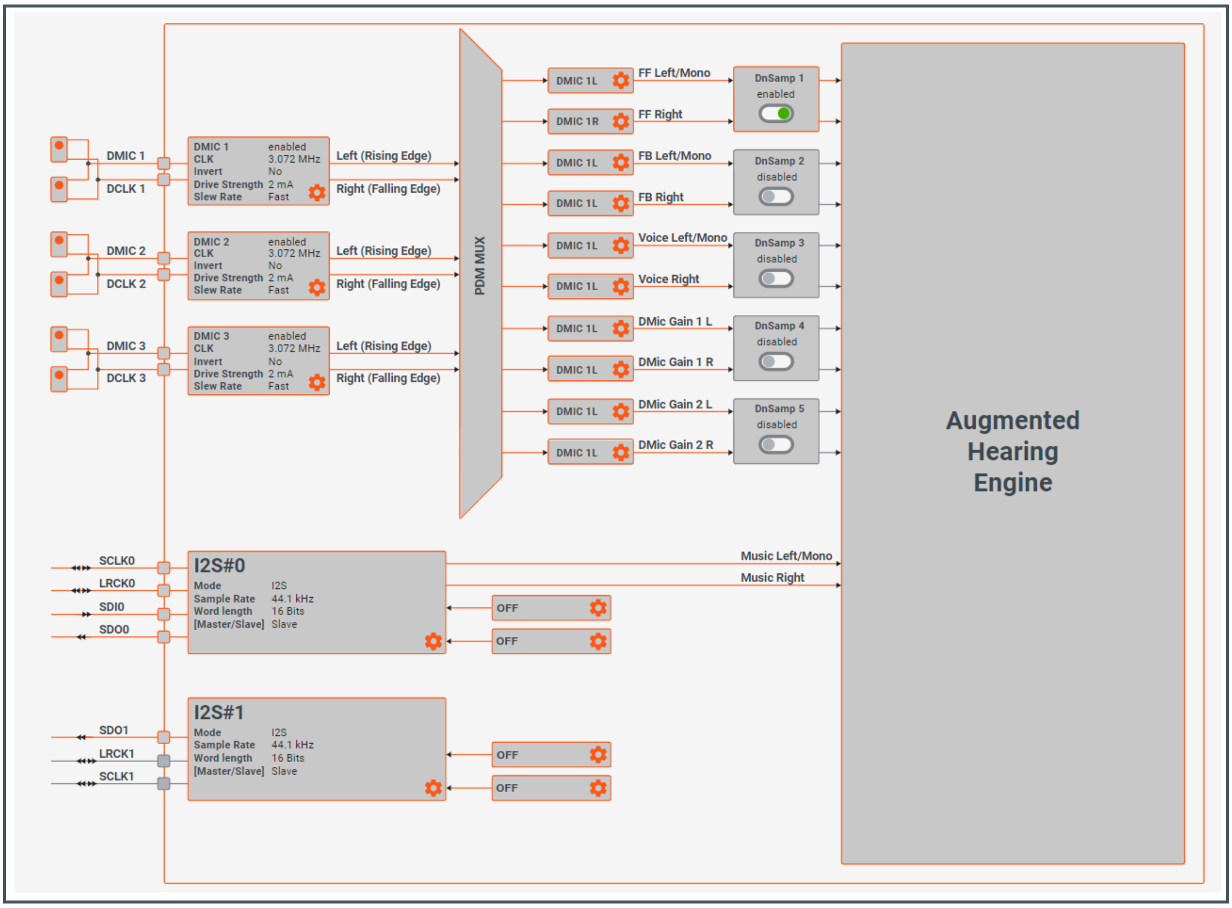
Figure 3:
Essential Connectors on the Evaluation Board



2.2 Project Setup

The first step when starting with a blank FleX Project is to make the necessary hardware configuration in the routing tab. Depending on the headphone's wiring, the correct PDM microphone must be routed to the corresponding filter. Here "DMIC 1L" refers to the first PDM interface, selecting on the rising clock edge, "DMIC1R" is the microphone synced on the falling edge.

Figure 4:
Minimal Routing Configuration



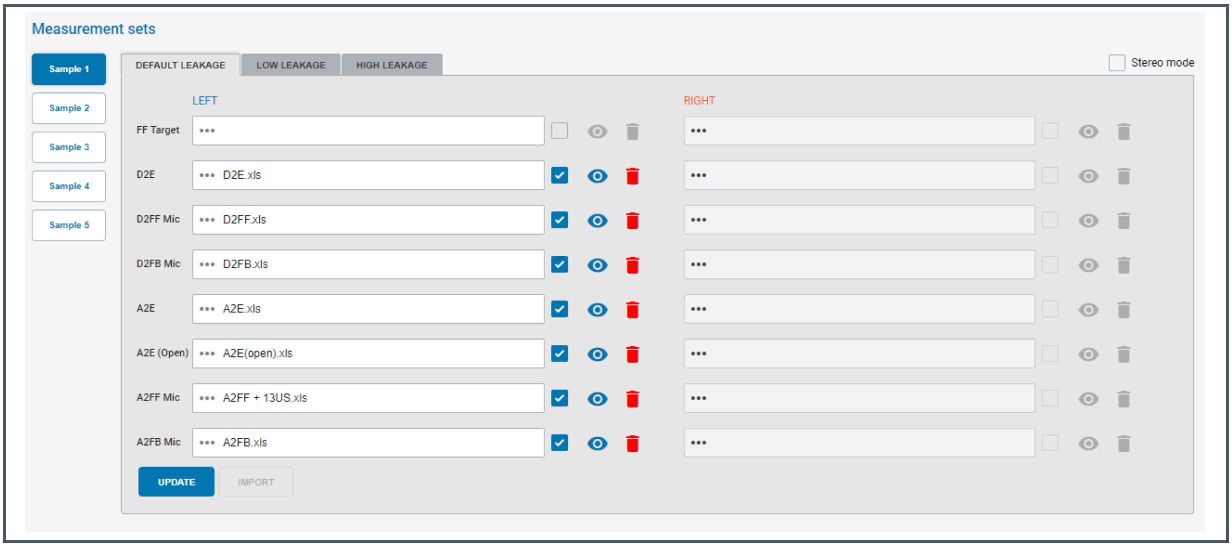
2.3 Measurements & Calibration

Next is loading the XLS files obtained from the characterization (for more details see appendix A on characterization) The AudioPrecision template provided by **ams** produces all the necessary files, with the same naming scheme used by FleX.

Characterization data for several samples can be loaded, to check how the filter performs on the various samples of the same headphone, thereby avoiding overfitting.

Load data by navigating to the project tab and select the required files by clicking on the dots symbol on each line.

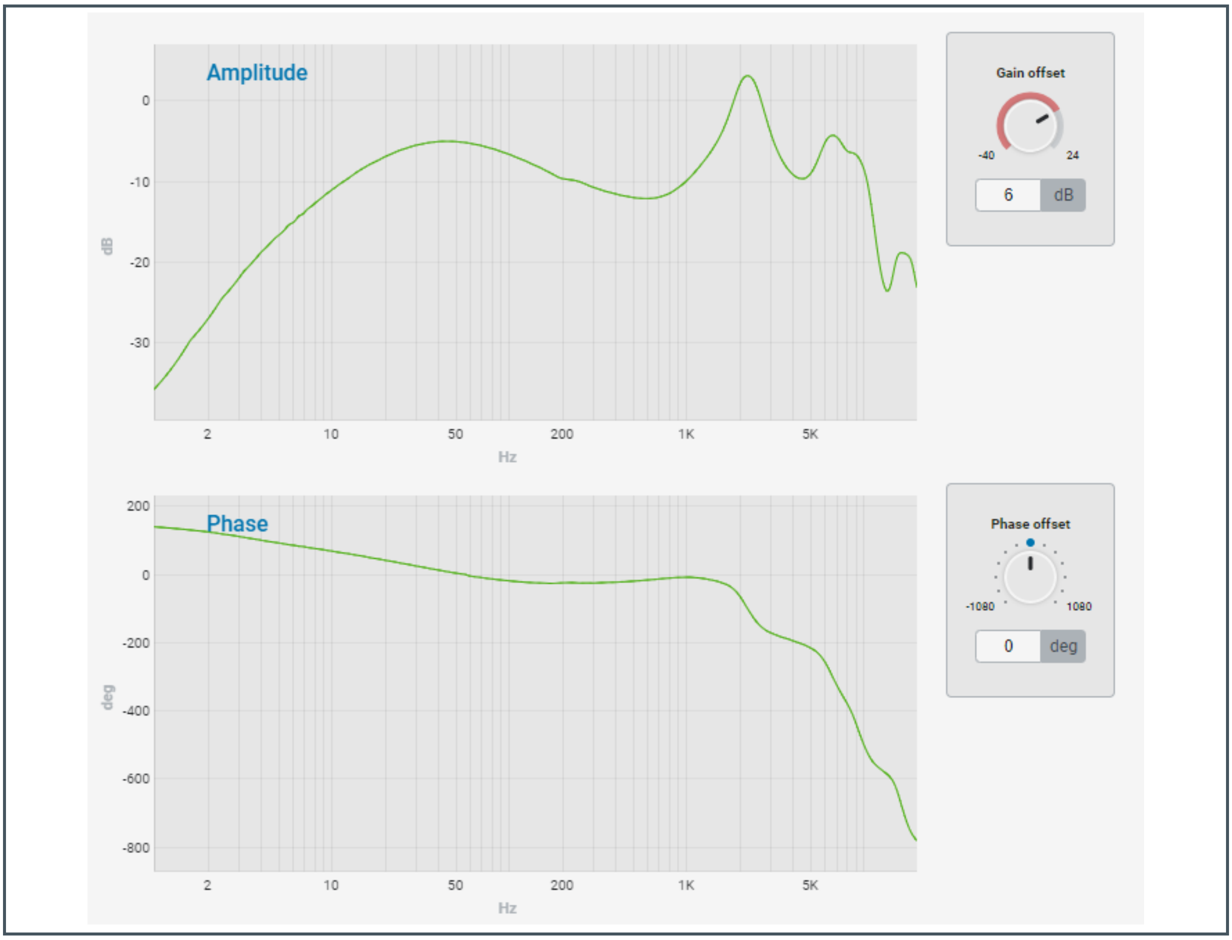
Figure 5:
Measurement Sets



Depending on the measurement setup it will be necessary to calibrate the measurements. To do so click on the eye symbol next to the affected measurement.

Example, if the measurement device’s output is balanced, but the headphone amplifier is connected single-ended, the signal will lose 6dB, which then should be compensated for:

Figure 6:
Gain and Phase Offset



If the headphone's speaker is wired inverse to the AS3460 than it was wired at the time of characterization it is necessary to compensate this using the "Phase offset". It is not necessary to calibrate for the test speaker or reference microphone, since their impact cancels out in the calculations.

2.4 Simulation Window

The simulation window consists of three plots; the first two comprise a Bode-plot. In the Bode-plot the ANC-filters and their target (or ideal Filter) can be viewed alongside each other. It is possible to **drag out the tuning tab** and view it parallel to the filter tuning tab. This way it is possible to tune filters and see the effect immediately. In the bottom plot FleX shows a prediction of the ANC-performance and the total noise reduction.

There are some basic controls for navigating the simulation window:

Figure 7:
Simulation Window Controls

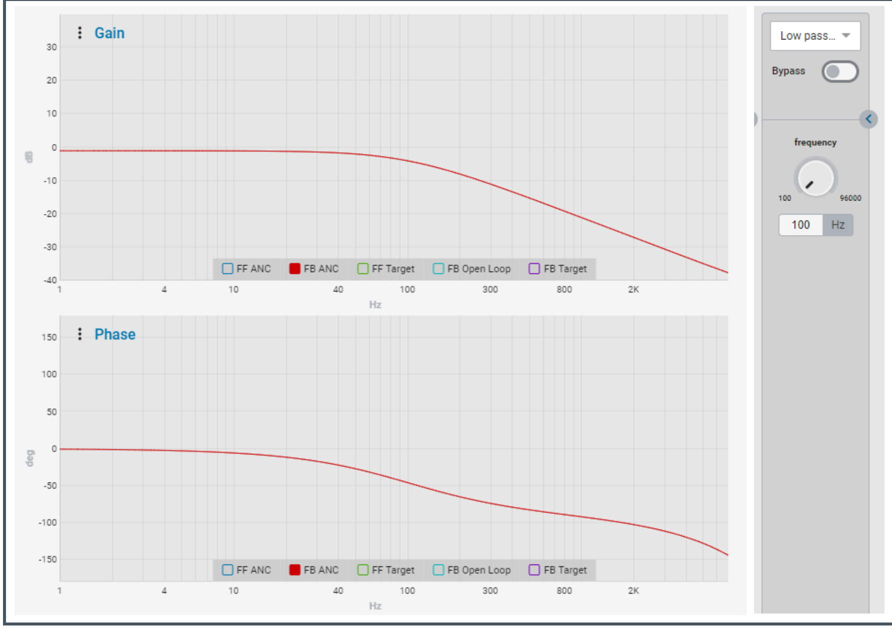
Command	Action
Shift + Mouse wheel	Zoom with respect to frequency
Ctrl + Mouse wheel	Zoom with respect to Y-axis
Double Click inside plot	Reset view
Click + Hold	Drag plot

2.5 Filter Tuning

For tuning both a feed forward and a feedback system, FleX offers intuitive tuning of predefined filters. For successful filter design it is crucial to be aware of how each filter type impacts both phase and magnitude. **The appendix contains a tutorial on how to tune the filters for feedback and feed forward ANC.**

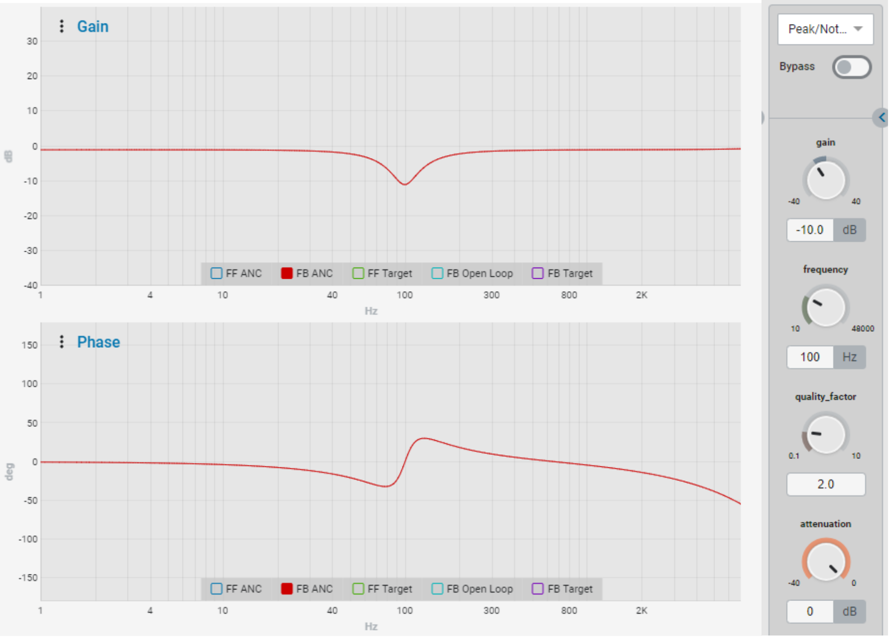
There are three main types of filters available:

Figure 8:
Low/High Pass



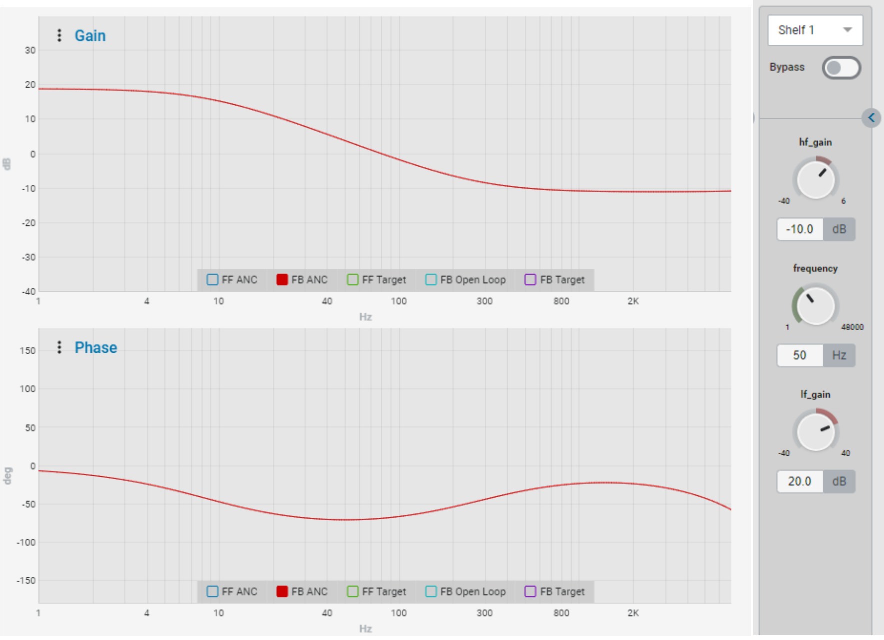
- Adjustable frequency
- Roll off of -6 dB/Oct.
- -45° phase shift at the roll off frequency

Figure 9:
Peak/Notch



- Adjustable frequency
- Gain
- And Q

Figure 10:
Shelf



- Adjustable frequency
- Separate gains for frequencies higher and lower than the center frequency
- Gain transition with 6dB/Oct.

Furthermore there is an Allpass filter and a 1st order x2 and you can also import biquad coefficients into FleX so you can load all types of filters with that.

A fixed number of instances of these filters can be used to build a filter. They can be controlled from the tuning tab.

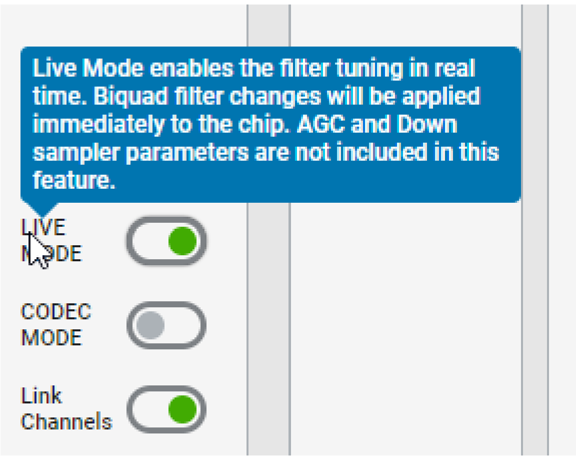
After completing a filter tuning, the filter can be activated on the AS3460 by clicking **“Write to AHE”**.

2.6 Tooltips

Within the FleX Software there is an extensive documentation of the features via Tooltips. Just hover over a function or button you want to know more about and the Tooltips are displayed immediately.

If the Tooltips functionality is not needed, you can also deactivate it via “Help – Disable Tooltips”.

Figure 11:
Tooltips Example



3 Evaluation Board

All block functions on the AS3460 Evaluation Board are explained in detail below.

3.1 Default Jumper Setting

The Evaluation Board has default jumper settings, which are marked in the color “Orange” below.

Figure 12:
Default Jumper Configuration

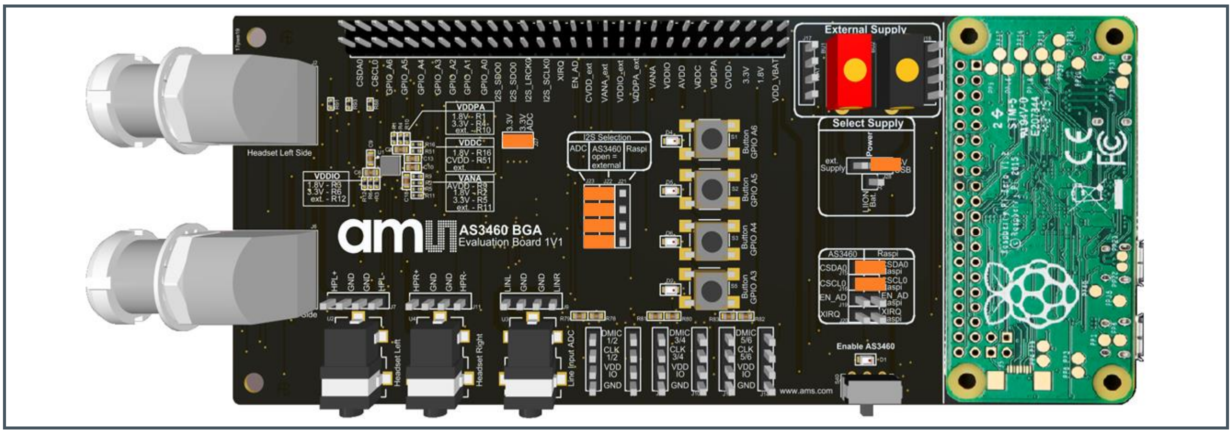


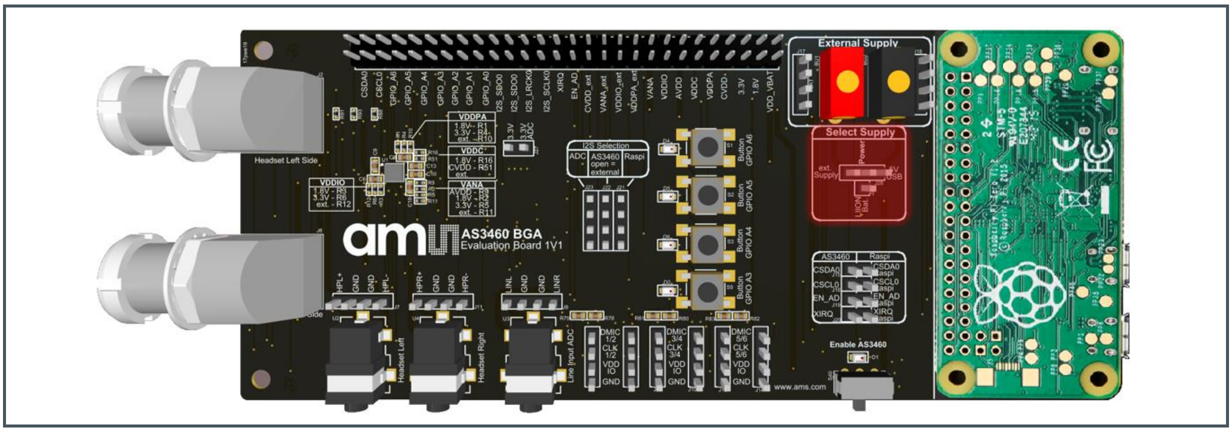
Figure 13:
Table - Default Jumper Configuration

Designation	Function	Conditions
J24	Select Supply	5 V USB to Power
J15	AS3460/Raspi	CSDA0 to CSDA0 Raspi
J16	AS3460/Raspi	CSCL0 to CSCL0 Raspi
J22/21	I^2S Selection	AS3460 Open = ADC
J28	3.3 V – 3.3 V ADC	ADC Supply

3.2 Power Supply

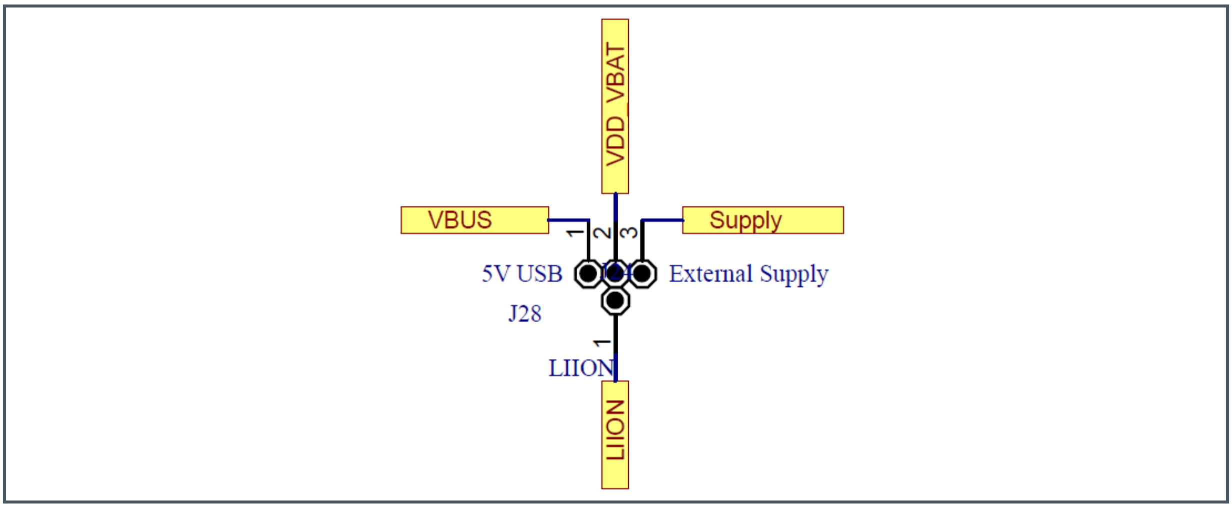
There are three different options to supply the hardware.

Figure 14:
Different Supply Options



The schematic drawing describes the different supply options, which can be chosen by setting the jumper J28 accordingly.

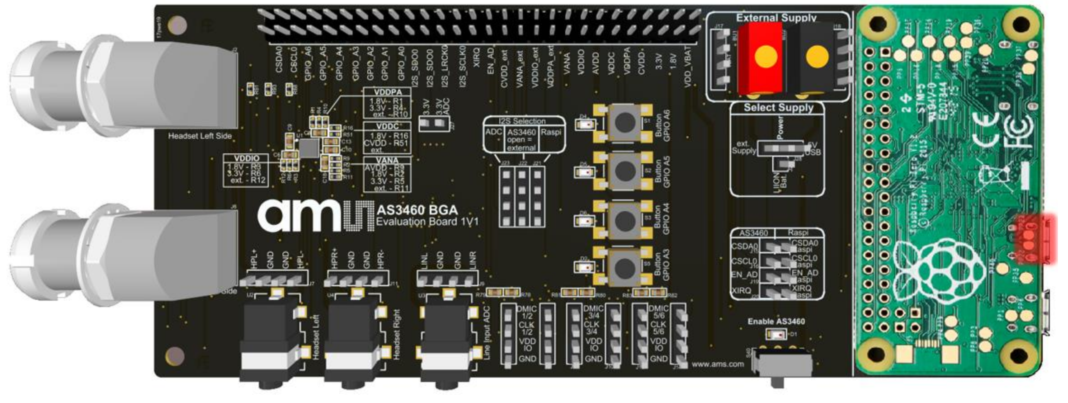
Figure 15:
Schematic - Different Supply Options



- Option 1: 5 V USB

Plug 5 V USB cable from the PC to the “Power 5 V- USB Connector” on the Raspberry Pi Board. Ensure that the USB port of the PC is USB 3.0. This setting will be the standard configuration setting on the board.

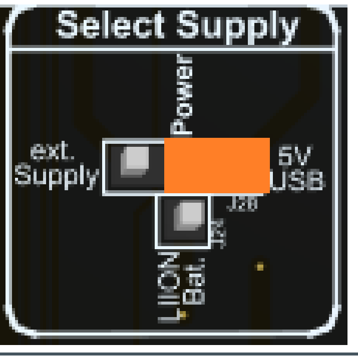
Figure 16:
Select Supply – Power Pin – 5 V USB



Therefore, set on the “Select Supply” header the **“Power Pin – 5V USB”**.

Please see jumper setting in Figure 17 below.

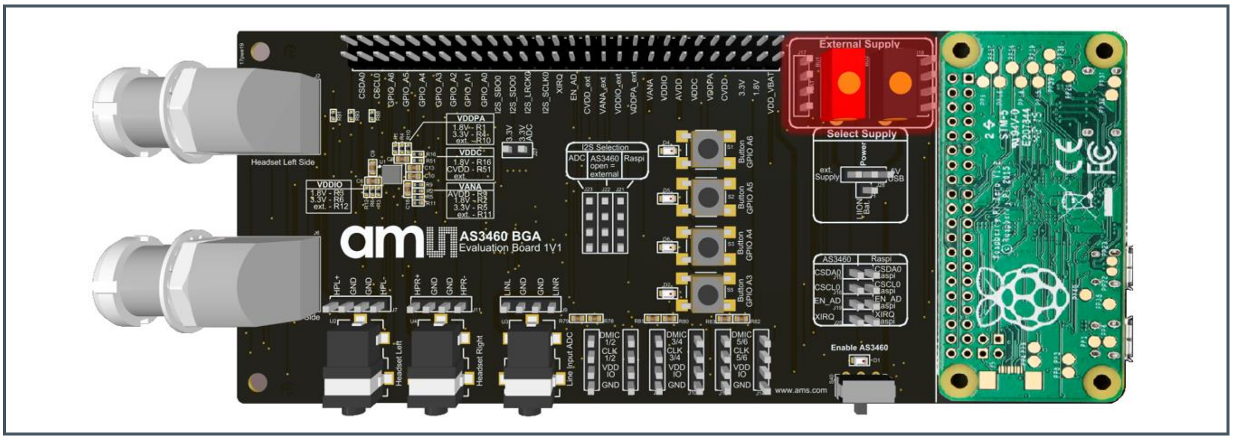
Figure 17:
Select Supply – Power Pin to “5 V USB”



- Option 2: External Supply (headers and connectors)

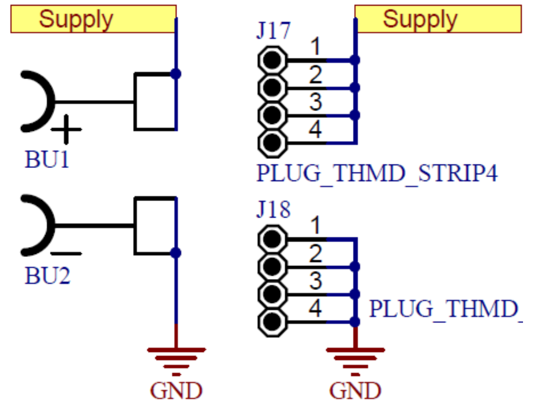
The Evaluation Board can be powered via an external power supply (3.3 V – 5 V) or with a Li-Ion-Battery, which can be connected to the “External Supply” headers or connectors.

Figure 18:
External Supply (headers and connectors)



The schematic below shows the external power supply connectors.

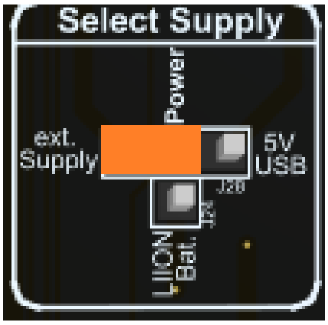
Figure 19:
Schematic - External Supply (headers and connectors)



Therefore, set on the “Select Supply” header the **“Power Pin – ext. Supply Pin”**.

Please see jumper setting in Figure 20 below.

Figure 20:
Select Supply – Power Pin to “ext. Supply”



- Option 3: Li-Ion Battery

The Evaluation Board can be powered via a Li-Ion-Battery, which can be soldered on the bottom side of the PCB. There are two solder pins, called “LI-ION-Battery Connection – Bat+1 / Bat-1” available.

Then the customer has a board, which works in standalone mode.

Figure 21:
LI-ION Battery Connection

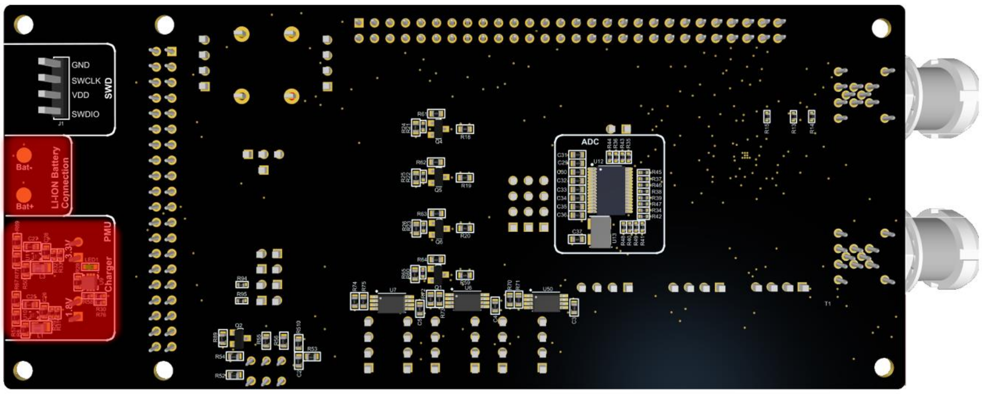
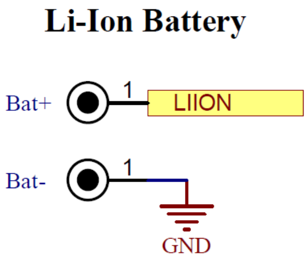


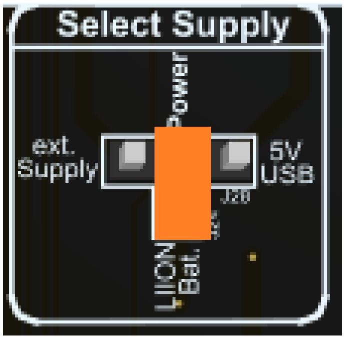
Figure 22:
LI-ION Battery - Schematic



Therefore, set on the "Select Supply" header the **"Power Pin – LI-ION Bat. Pin"**.

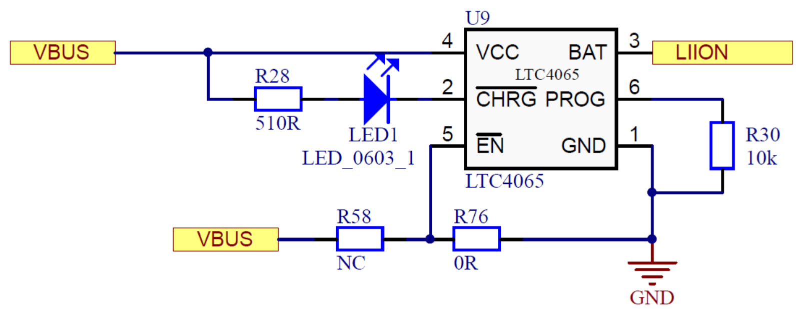
Please see jumper setting in Figure 23 below.

Figure 23:
Select Supply – Power Pin to "LI-ION Bat"



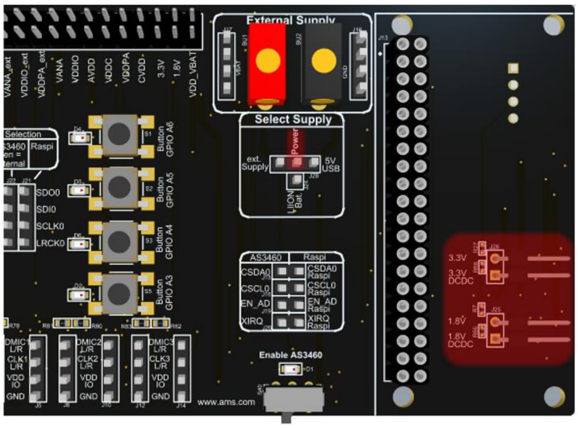
At the PMU (Power Management Unit) section, there is a charger assembled, which provides the proper supply for the battery.

Figure 24:
Charger – Schematic



In addition there are DCDCs (3V3, 1V8) assembled, which provide the power supply for the AS3460 IC.

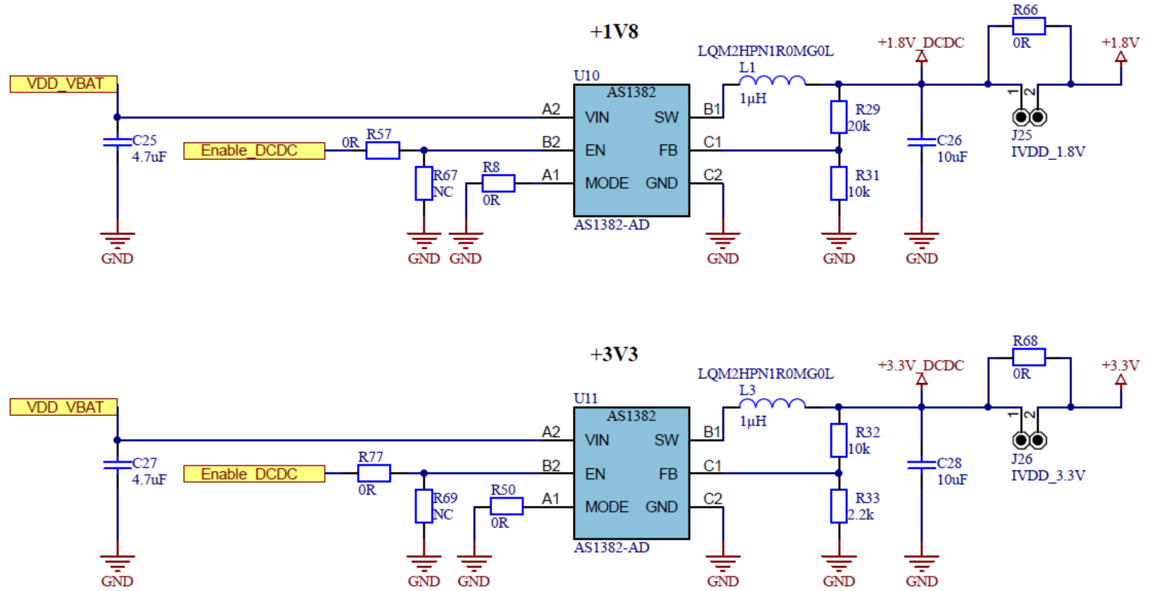
Figure 25:
Measurement Pins – Power Pin and DCDCs



There are also measurement pins of the DCDC (below the Raspberry Pi PCB)

Power Pin 3.3 V DCDC to 3.3V and **Power Pin** 1.8 V DCDC to 1.8 V.

Figure 26:
DCDC Schematic

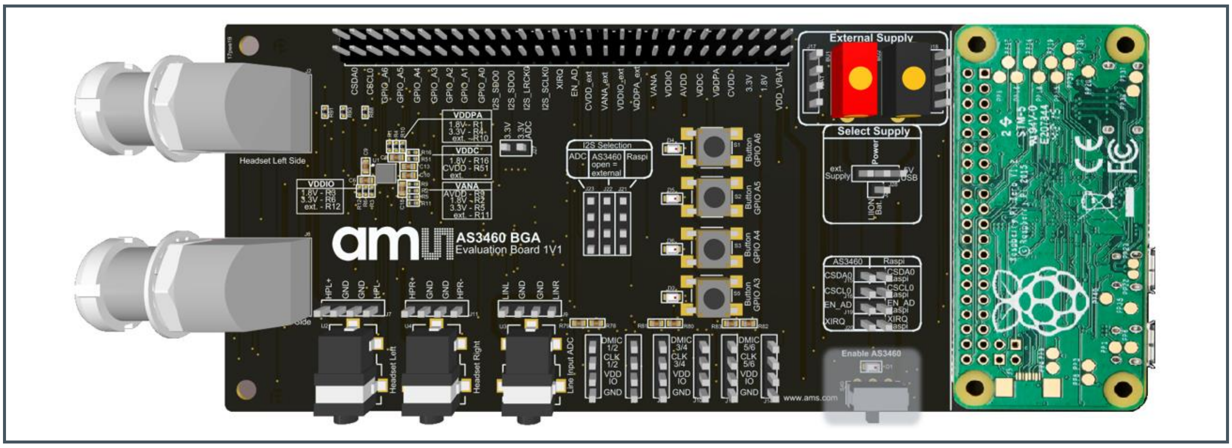


3.3 Power ON/OFF

The device start-up and power-down is controlled with the EN pin of AS3460. A rising edge on the EN input pin will trigger start-up of the device. Once EN pin is high the integrated regulators are powering up to generate analog audio- and digital core supply voltage. The EN pin of AS3460 is typically controlled with a GPIO of the host device.
Power down of AS3460 is triggered by pulling EN pin low for minimum time. Once this minimum time is reached, the power down sequence is initiated and no more I²C commands from the host device are accepted. The chip fades out ANC and audio playback to avoid pop noise when shutting down the device completely.

Connect the power supply to the board (choose one of the three options mentioned above).
To enable the AS3460 on the evaluation board, the "Enable AS3460" switch has to be powered/switched on. If the AS3460 is powered, the LED above the switch is on.

Figure 27:
Power ON/OFF Switch



3.4 Headphone Connectors

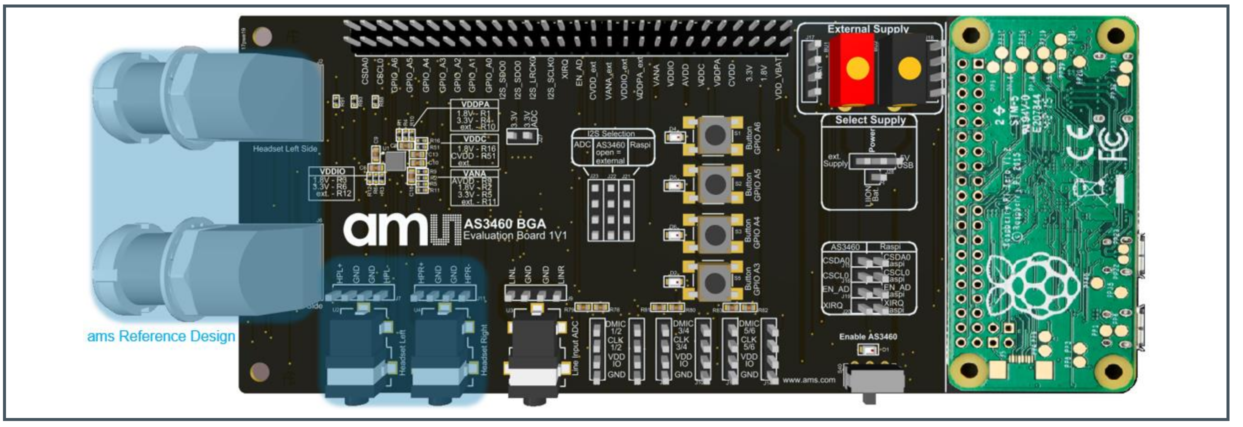
The AS3460 features two fully differential stereo headphone amplifier outputs, which are suitable for direct connection to external speakers. Each channel is designed to support speaker loads of typically 32 Ω down to 16 Ω.

With a supply voltage (AVDD) of 1.5 V the amplifier is capable of delivering an output power of typ. 26 mW into a 32 Ω load. In case a 16 Ω speaker is connected the maximum possible output power can be increased to 52 mW per channel.

Headphone connectors (3.5 mm audio jack connectors and headers) for the left and the right headset side are available.

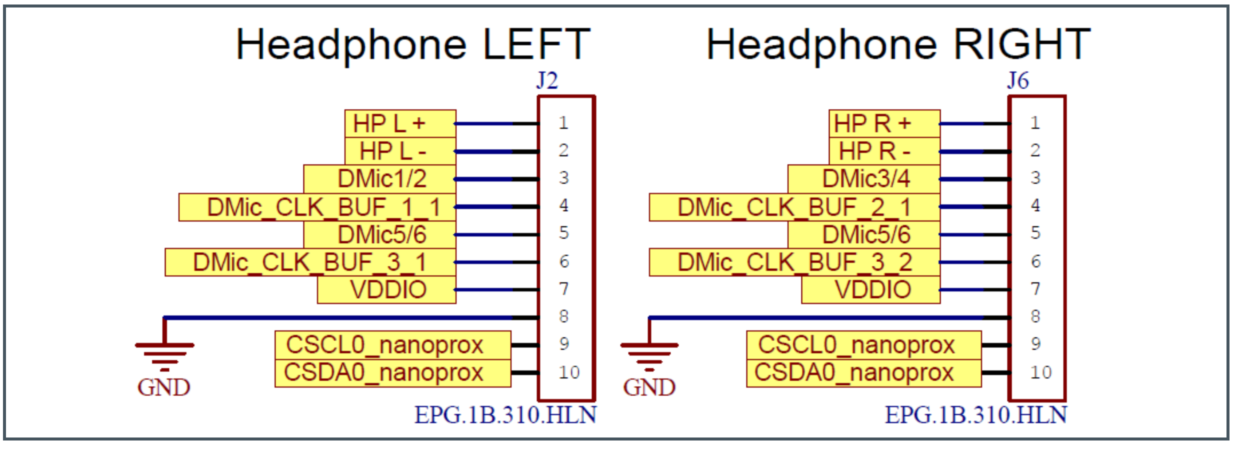
In general, the LEMO connectors are not assembled on the evaluation board, but the footprints are available on the board. These LEMO connectors are used for all **ams** reference designs (reference earbud demonstrators).

Figure 28:
Headphone Connectors



The LEMO connectors can be soldered on the evaluation board. For soldering, it is mandatory to follow the pinout according to the evaluation board schematic below.

Figure 29:
LEMO Connector Pinout

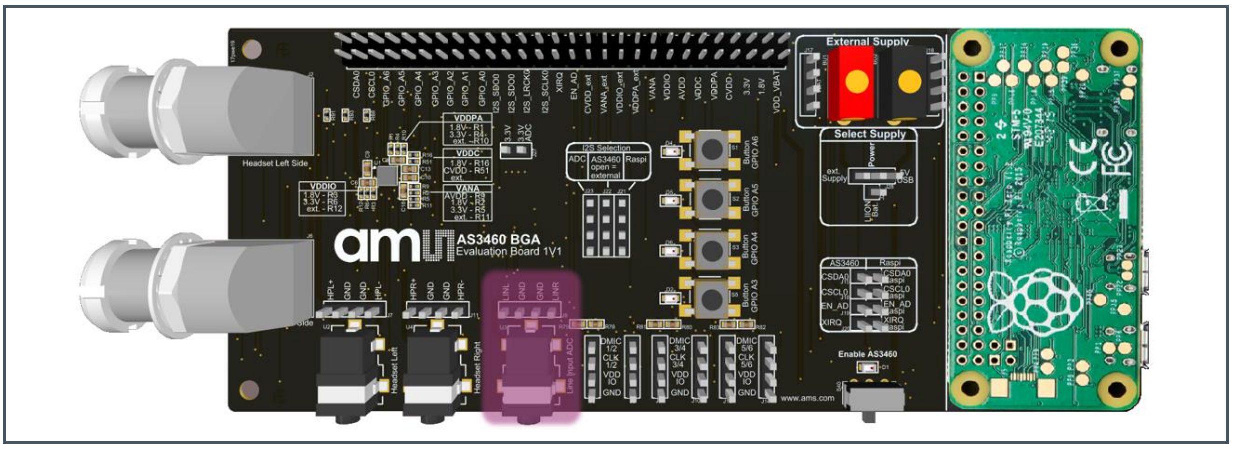


Connectors can be found on the LEMO homepage: www.lemo.com/pdf/EPG.1B.310.HLN.pdf

3.5 Line Input Connectors

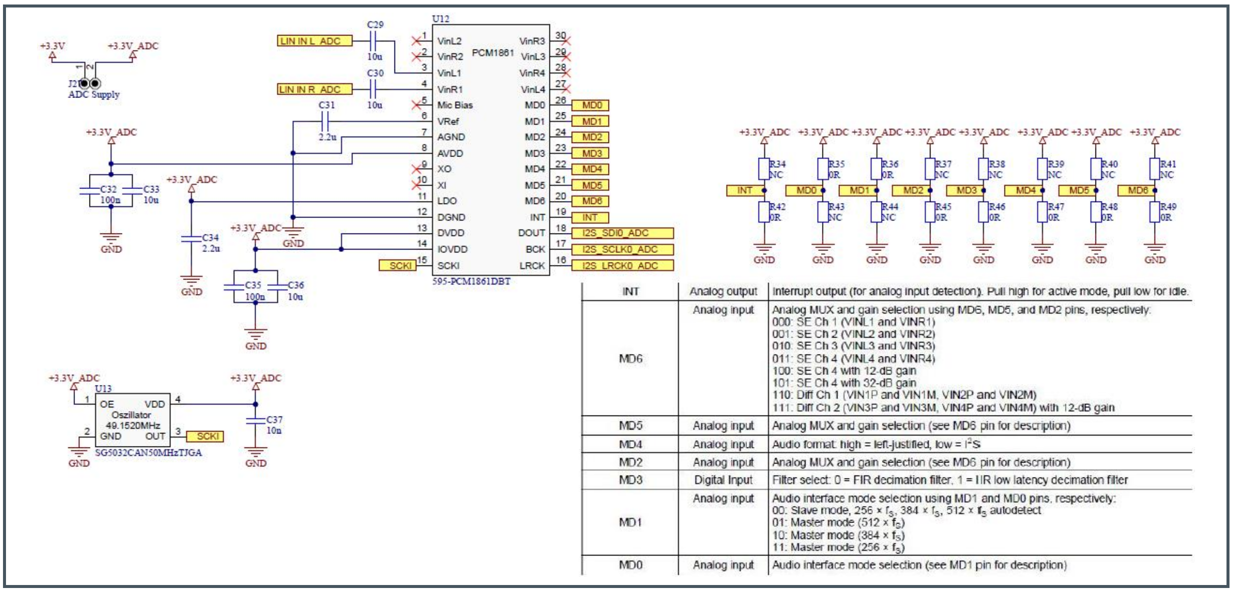
The AS3460 does support one stereo I²S input and two stereo output channels. In addition, the AS3460 Evaluation Board has a Line Input ADC Connector (3.5 mm audio jack connectors and headers) for music playback.

Figure 30:
Line Input Connector



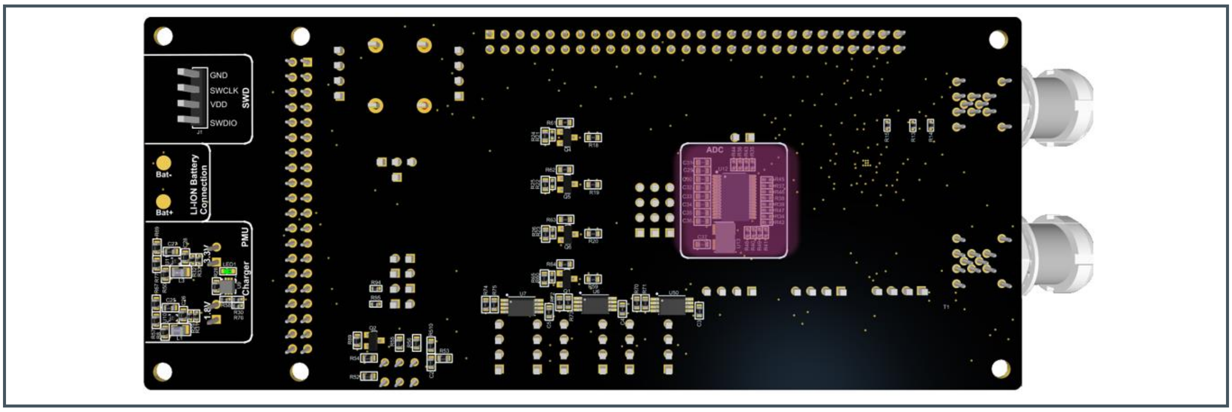
If the user wants to play music through the line input to the headphone output, use the assembled ADC on the bottom side of the board.

Figure 31:
ADC - Schematic



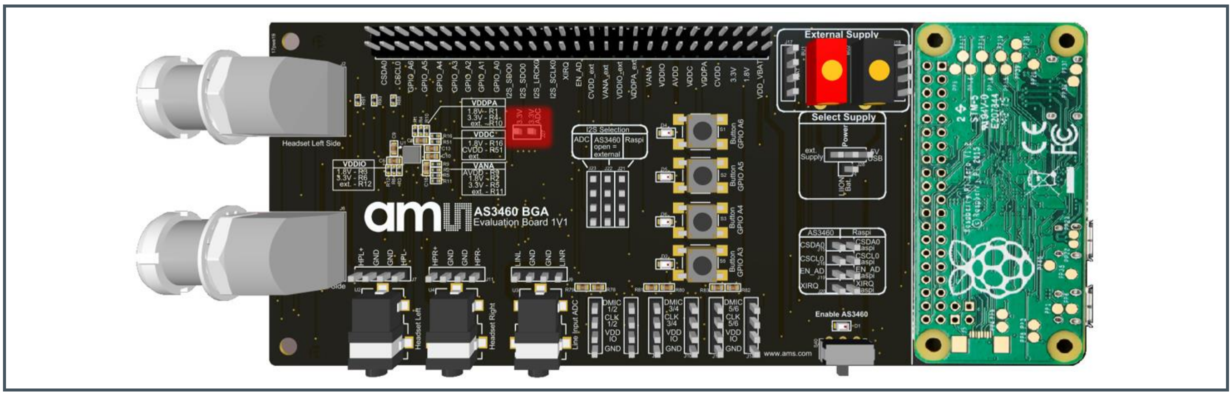
INT	Analog output	Interrupt output (for analog input detection). Pull high for active mode, pull low for idle.
MD6	Analog input	Analog MUX and gain selection using MD6, MD5, and MD2 pins, respectively: 000: SE Ch 1 (VINL1 and VINR1) 001: SE Ch 2 (VINL2 and VINR2) 010: SE Ch 3 (VINL3 and VINR3) 011: SE Ch 4 (VINL4 and VINR4) 100: SE Ch 4 with 12-dB gain 101: SE Ch 4 with 32-dB gain 110: Diff Ch 1 (VIN1P and VIN1M, VIN2P and VIN2M) 111: Diff Ch 2 (VIN3P and VIN3M, VIN4P and VIN4M) with 12-dB gain
MD5	Analog input	Analog MUX and gain selection (see MD6 pin for description)
MD4	Analog input	Audio format: high = left-justified, low = I²S
MD2	Analog input	Analog MUX and gain selection (see MD6 pin for description)
MD3	Digital Input	Filter select: 0 = FIR decimation filter, 1 = IIR low latency decimation filter
MD1	Analog input	Audio interface mode selection using MD1 and MD0 pins, respectively: 00: Slave mode, 256 × f_S, 384 × f_S, 512 × f_S autodetect 01: Master mode (512 × f_S) 10: Master mode (384 × f_S) 11: Master mode (256 × f_S)
MD0	Analog input	Audio interface mode selection (see MD1 pin for description)

Figure 32:
ADC - PCB Bottom Side



Therefore, set jumper J27 (3.3 V – 3.3 V ADC) to supply the ADC.

Figure 33:
Supply for ADC IC



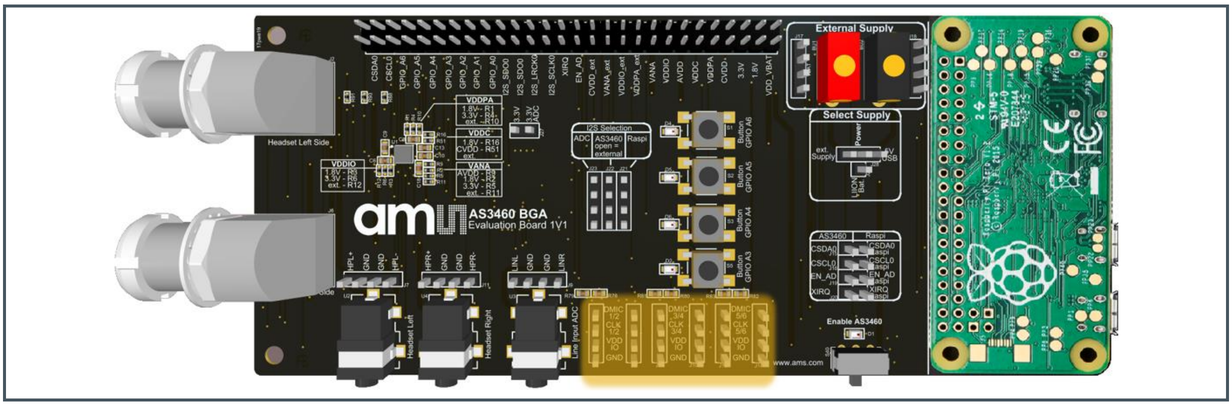
3.6 Digital Microphone Connectors

The AS3460 supports three stereo PDM digital microphone interfaces. Each clock interface features a dedicated clock output, which can be controlled during operation. This allows for power optimization in case in certain operation modes of the device the microphone is not needed. Furthermore, it is possible to invert the clock output to improve the EMI interference in an application.

The integrated oscillator and PLL do provide the system clock for AS3460, which is also used for the digital microphone interface. It is possible to operate the microphone interfaces with different clock rates, which may help to reduce power consumption in case the full bandwidth of the microphones is not required.

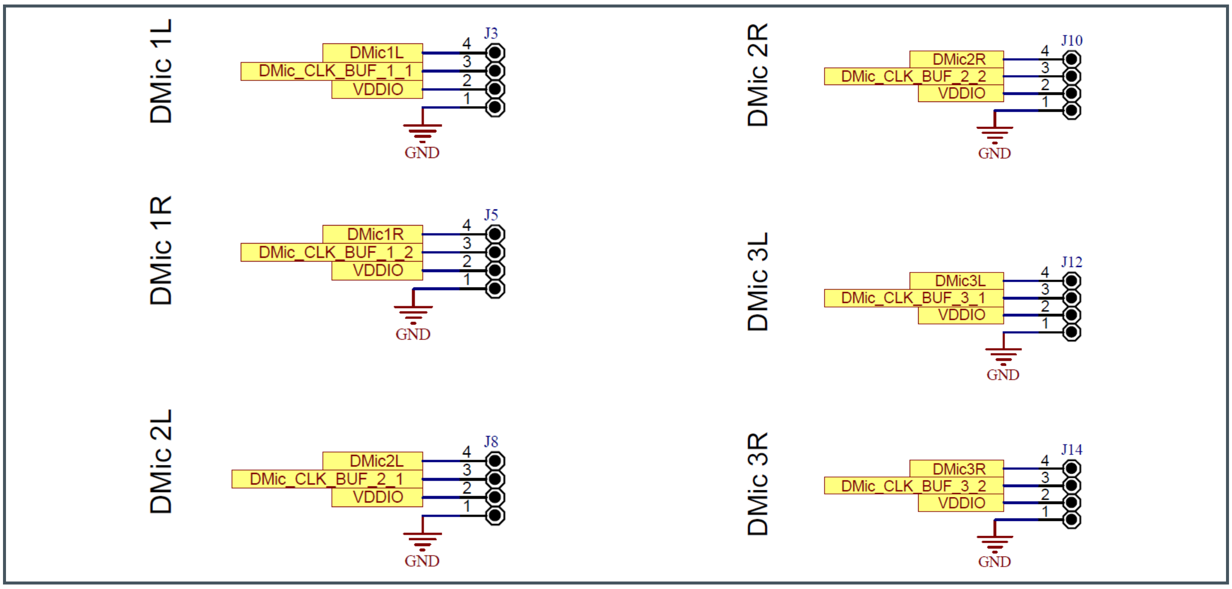
Besides the given flexibility of the different clock output configurations AS3460 features also flexible microphone signal routing to the AHE. However, there is no particular rule for the microphone assignment to the microphone inputs. The device has internal PDM data multiplexers that allow the assignment of any microphone data channel to any AHE data input. Furthermore additional multiplexers are available to feed back any microphone signal from any input channel to the I2S#0 and I2S#1 interface.

Figure 34:
Digital Microphone Headers



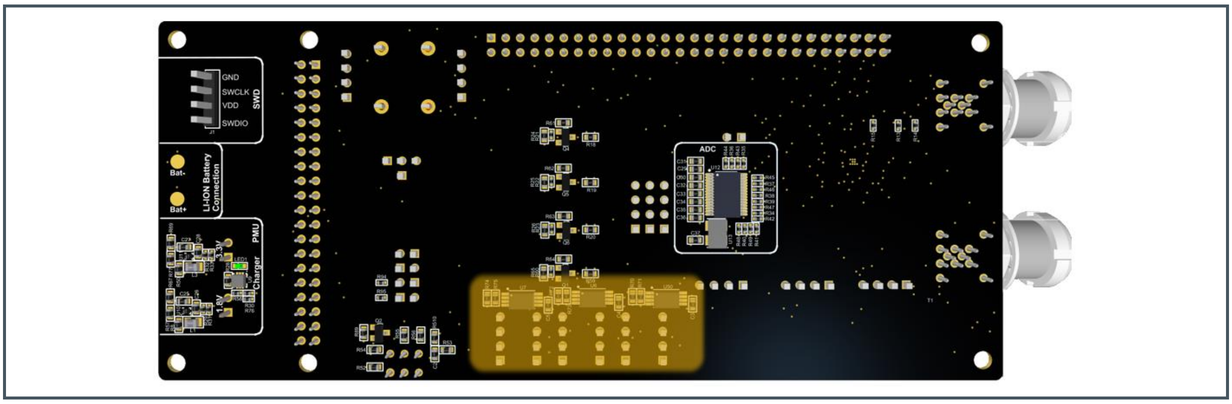
The schematic shows the pinout of the digital microphone headers.

Figure 35:
Digital Microphone Headers - Pinout



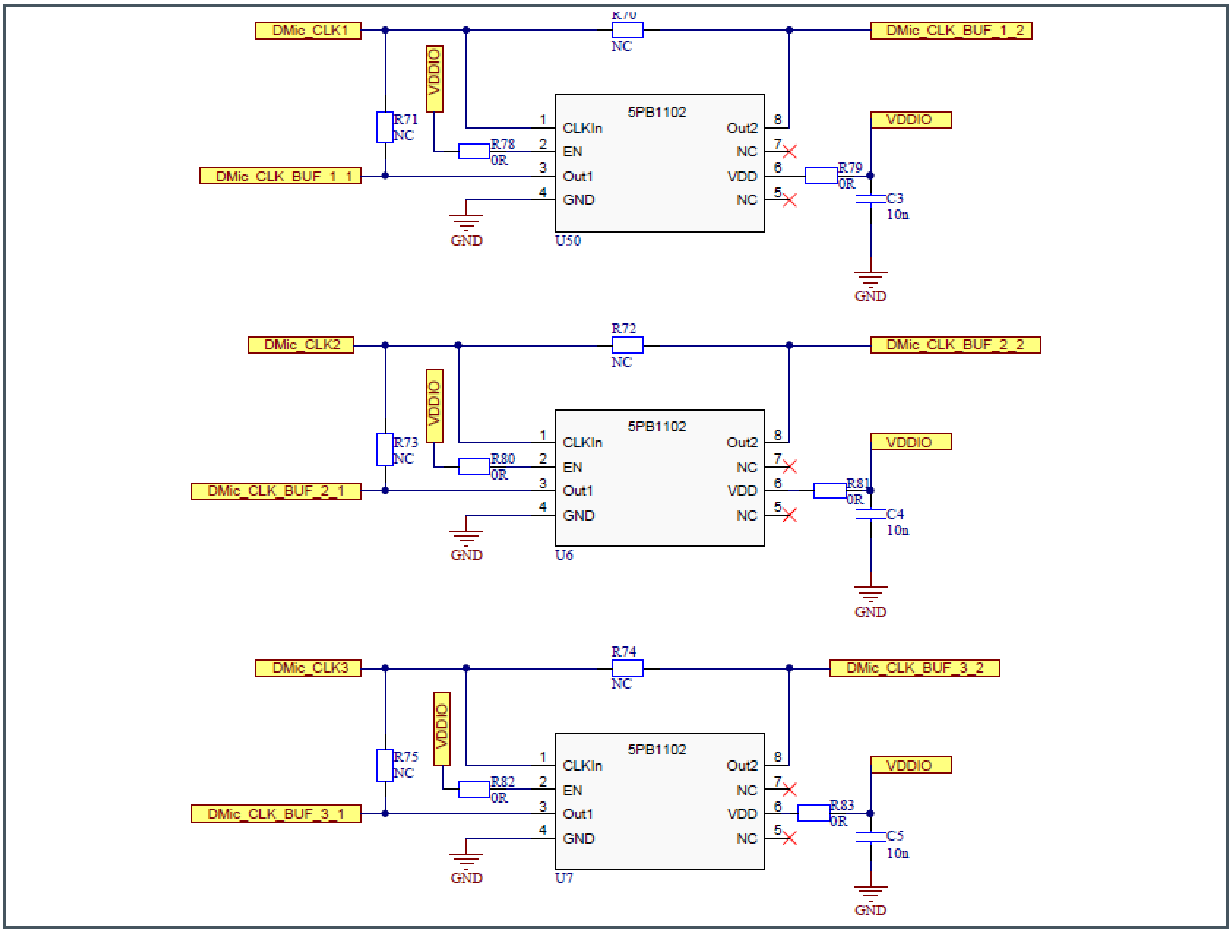
On the bottom side of the AS3460 Evaluation Board are microphone clock buffers assembled, which support headset with longer cables with a proper clock signal.

Figure 36:
Clock Buffers



The schematic below shows the used buffers and the pin configuration of these parts.

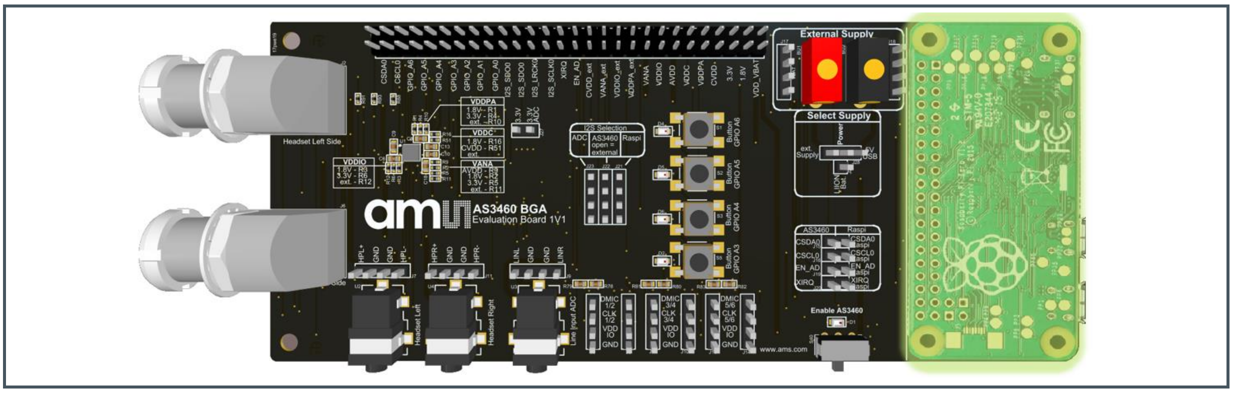
Figure 37:
Schematic - Clock Buffers



3.7 Raspberry Pi Board

The evaluation board has a Raspberry Pi Board (Raspi) hocked on. The Raspberry Pi Board can be used on the one hand as power supply and on the other hand optionally as I²C Interface.

Figure 38:
Raspberry Pi Board



3.8 I²C Communication

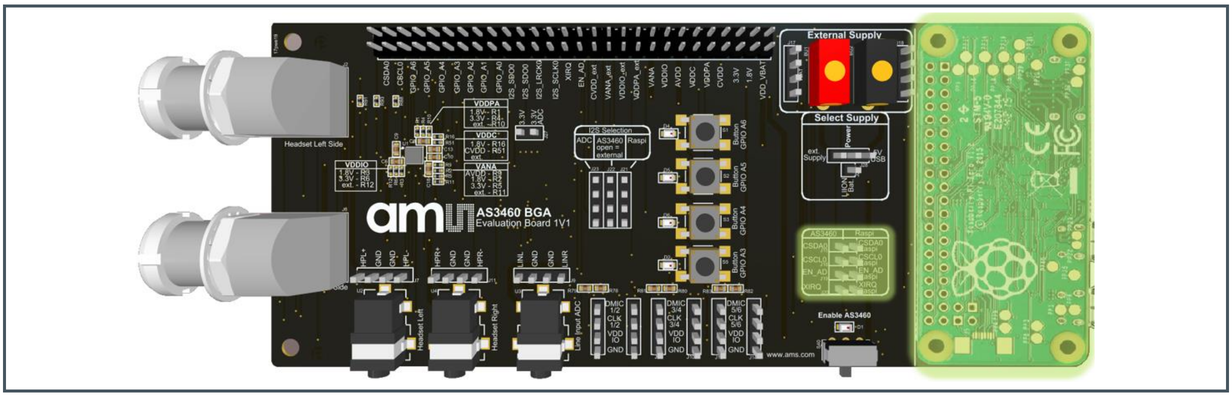
AS3460 features an I²C Slave interface, which is the primary control interface to control chip functionality like operation modes. Furthermore, the interface is used to store and modify AHE program code as well as system configuration settings, which are application dependent and not part of the device firmware. The interface requires two pull up resistors for proper operation. The two interface pins SCL and SDA do support a wide supply voltage range from typical 1.8 V up to maximum 3.6 V. The interface is supplied via VDDIO pin and all voltage thresholds are derived from V_{VDDIO} voltage.

The default I²C address for AS3460 is 0x45. This can be changed to 0x44 by pulling GPIO A0 high and power cycling the device. This is illustrated in Figure 103 for the lower board. It is necessary to restart FleX whenever the I²C address of AS3460 is changed.

Note that however GPIO A0 is configured in FleX, the functionality to select the I²C address will always occur. This means that when a different function is programmed on GPIO A0, care must be taken to ensure the correct level on GPIO A0 at boot time.

I²C Interface connection between Raspberry Pi Board and AS3460 is the highlighted “AS3460/Raspi” header. This is the standard connection to the PC to communicate with the FleX Filter Design Software.

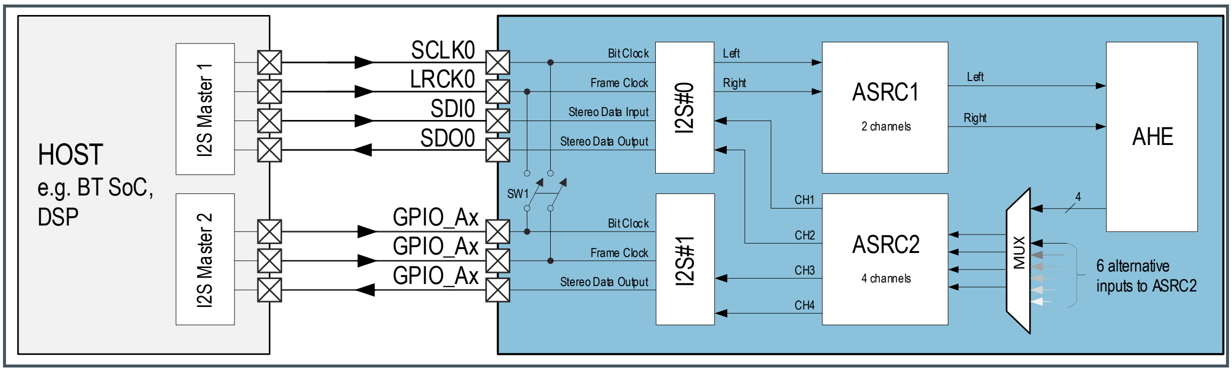
Figure 39:
Raspberry Pi Board Used For I^{2}C Interface



3.9 I^{2}S Selection

The AS3460 does support one stereo I^{2}S input and two stereo output channels. The primary I^{2}S interface I2S#0 (SCLK0, LRCK0, SDI0 and SDO0) is used to receive audio data from an I^{2}S master host device which can be for example a BT SoC or Digital Signal processor. The received stereo audio data is fed into an Asynchronous Sample Rate Converter (ASRC1) to synchronize it to the internal clock domain of AS3460. The output of ASRC is fed into the AHE for additional signal processing and audio playback. The stereo audio output (SDO0) at I2S#0 interface allows to send for example raw microphone data back to the host. This function allows sharing for example the ANC microphones to be used as voice call microphones. In order to be synchronous to the host the audio data is synchronized via a second four channel ASRC2 to the primary I^{2}S interface.

Figure 40:
I^{2}S Interface – Block Diagram



The second I^{2}S interface I2S#1 does support one stereo output only. This allows sending out two additional audio channels back to the host to apply algorithms or further enhance the audio signal. I2S#1 interface receives its data from ASRC2, which can support up to six different input channels for highest system flexibility when developing software algorithms on the host system making use of the

ANC microphones. In order to minimize external PCB routing effort when both stereo outputs are in use it is possible via internal switch SW1 to route SCLK0 and LRCK0 signal to I2S#1. If this connection is enabled, there is just a single additional signal line necessary to transfer the four audio channels back to the host.

The digital audio interface is compliant to the following audio standards:

- Left Justified
- I²S

The I²S connection (highlighted “I²S Selection”) gives the option to connect the I²S pins of the AS3460 to the ADC (for music playback) or to the Raspberry Pi Board.

Figure 41:
I²S Selection Pins

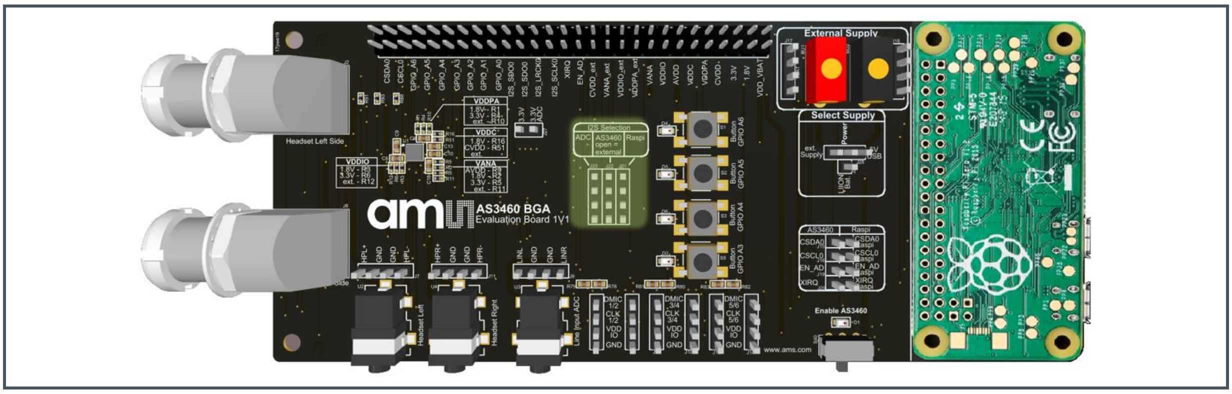


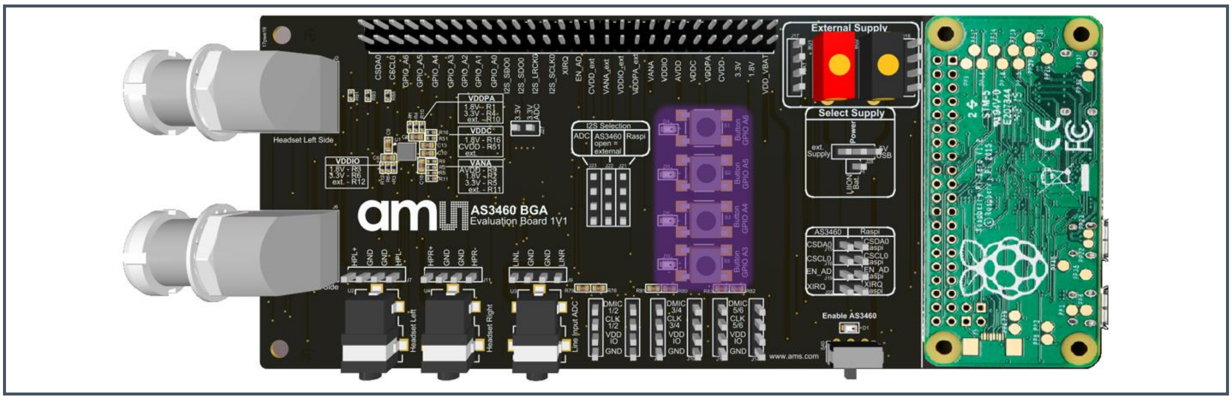
Figure 42:
Table - I²S Selection Pins

ADC	AS3460	Raspi
-	SDO0	SDO0_Raspi
SDIO_ADC	SDIO	SDIO_Raspi
SCLK_ADC	SCLK	SCLK_Raspi
LRCLK_ADC	LRCLK	LRCLK_Raspi

3.10 GPIO Buttons

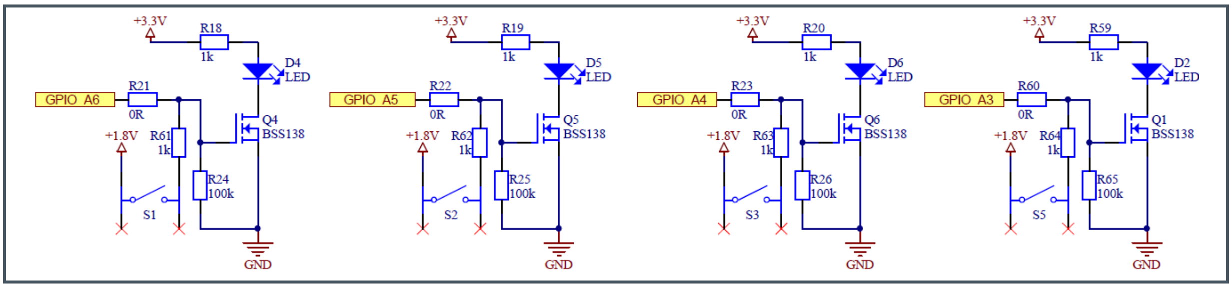
AS3460 supports a general-purpose I/O interface, which allows to be configured as button interface for standalone mode operation to switch between ANC and augmented operation modes or mute audio outputs. When configured as inputs each pin can have internal pull-up/down resistor enabled to avoid floating inputs.

Figure 43:
GPIO Pins Used as Buttons Function



On the Evaluation Board GPIO A3 pin to GPIO A6 pin connected to buttons as shown in the Figure 44

Figure 44:
GPIO Pins (A6 to A3) Used as Buttons Function - Schematic

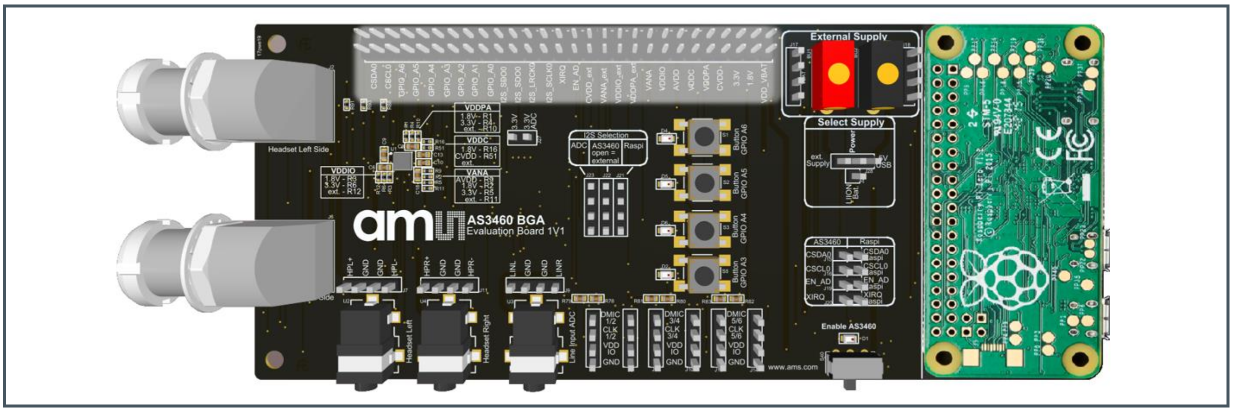


Besides the button, input function an I²S stereo output can also be mapped to any of the GPIO pins of AS3460 Please also mind that due to the integrated GPIO multiplexer any button or I²S function can be mapped to any of the GPIO pins.

3.11 Measurement Header

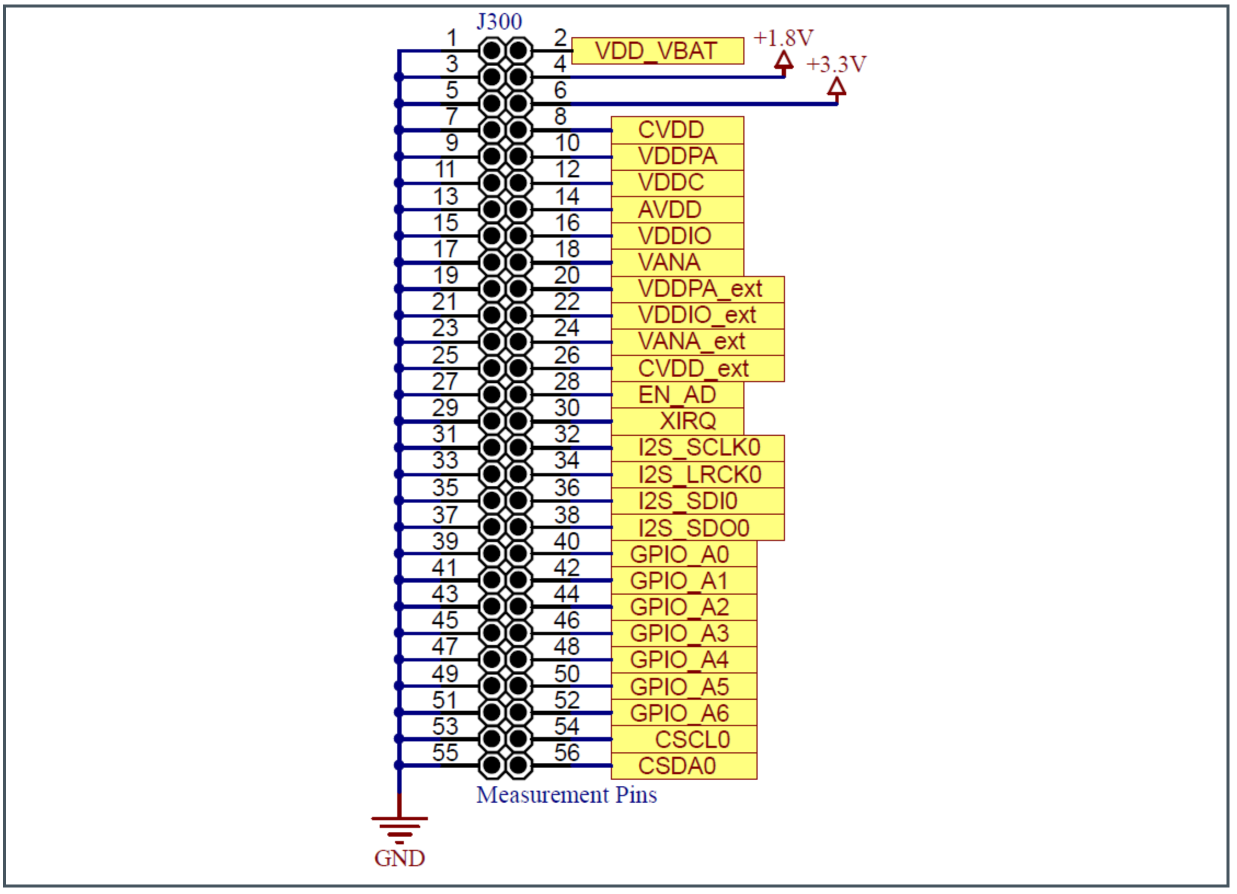
This header provides every pin of the AS3460 IC, the power and DCDC supplies. The second pin row is ground. The measurement pins can also be used to connect for instance a BT SoC to the evaluation board to test the hardware.

Figure 45:
Measurement Header



Pinout of the measurement header J300 is shown below.

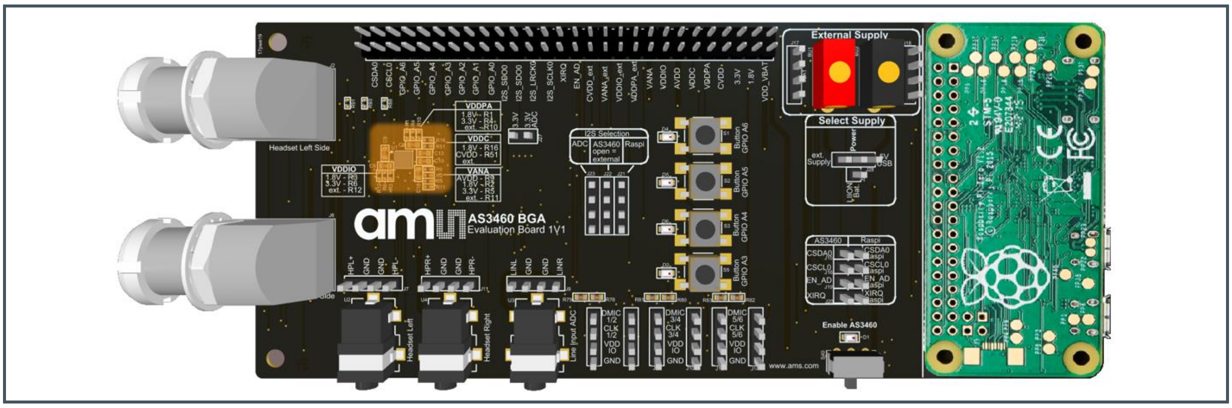
Figure 46:
Measurement Header – Pinout



3.12 AS3460 IC

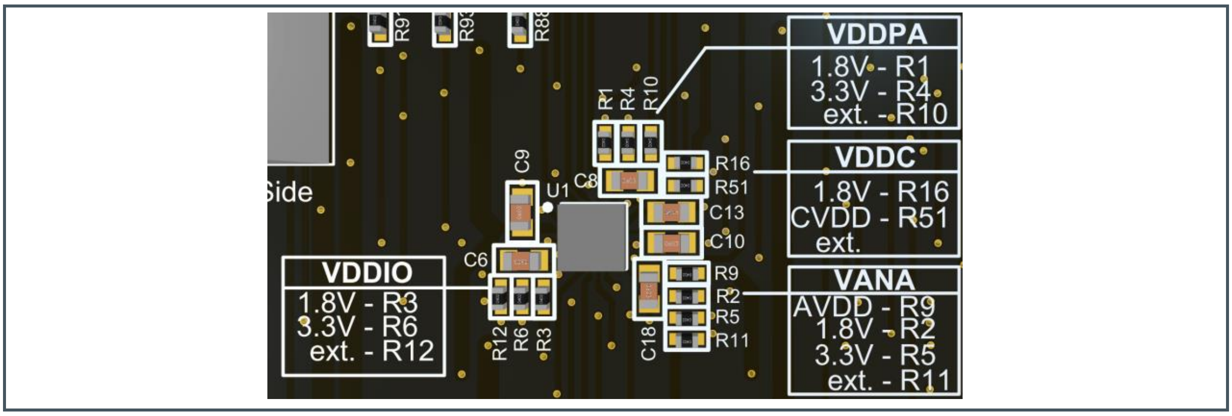
The AS3460 is a digital augmented hearing companion device. The figure below shows the AS3460 IC and the external components on the Evaluation Board.

Figure 47:
AS3460 IC with External Components on Evaluation Board



The zoom in picture displays the AS3460 IC with the passive components (capacitors and resistors) around in detail.

Figure 48:
Zoom in - AS3460 IC with External Components



In real application, the digital augmented hearing device just needs seven external components to operate:

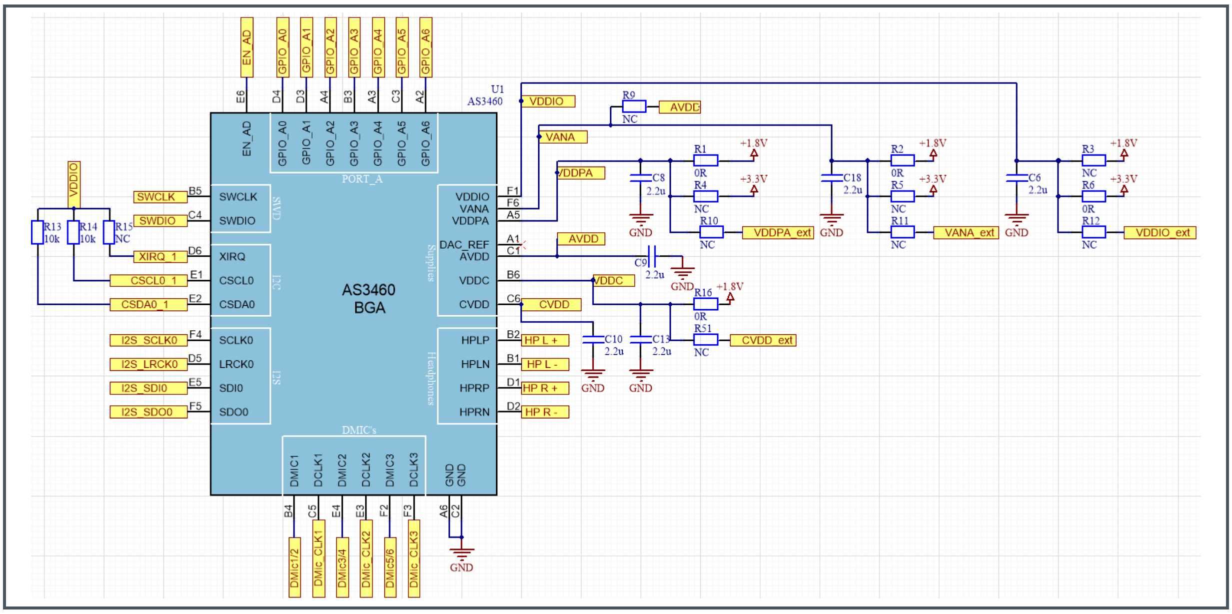
- 4 x 1 µF capacitor
- 1 x 100 nF capacitor
- 2 x 10 k pull-up resistors
- No external crystal necessary!
- No external flash necessary!

In addition to the necessary coupling capacitors and the I^{2}C pull-up resistors, the user can configure individually the device voltages according to their application. This configuration can be done via soldering the 0R resistors to the corresponding footprint.

The standard configuration of this Evaluation Board is the following:

- VDDPA: 1.8 V – R1
- VDDC: 1.8 V – R16
- VANA: 1.8 V – R2
- VDDIO: 1.8 V – R6

Figure 49:
Schematic – Different Configurations of the Supply Pins of AS3460

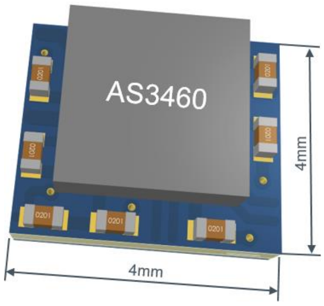


Every supply pin can be connect to +1.8 V, +3.3 V or to an external supply. Therefore, the correct resistor has to be assembled on the board. The external supply has to be connected to the correct pin on the measurement header J300.

4 PCB Design

The figure represents the necessary amount of external components.

Figure 50:
AS3460 IC with the Necessary Amount of External Components



This table reflects the PCB size of a single sided PCB.

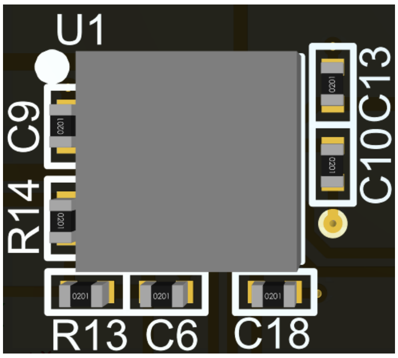
Figure 51:
Table - PCB Area Calculation/Single Sided PCB

Component	Description	Component Count	Footprint [INCH]	PCB Area Total [mm^2]
AS3460	ANC companion device	1	AS3460_BGA36	10.240
4.7 k resistors	I^{2}C pull-up resistors	2	R0201_S	1.210
100 nF capacitors	VDDIO input blocking capacitors	1	C0201_S	0.605
2.2 µF capacitors	AVDD, VANA, VDDC, CVDD LDO input/output capacitors	4	C0201_S	2.419
Total net area				14.474
Total gross area incl. 10% routing overhead				15.921

4.1 PCB Layout Recommendation

The Printed Circuit Board (PCB) layout recommendation for the AS3460 (shown in Figure 50 and Figure 52) is given below:

Figure 52:
PCB Area Calculation / Single Sided PCB



It is recommended to use a four Layer PCB to route all pins of the AS3460 IC.

4.1.1 Component Placement Recommendation

This chapter provides detailed information about recommended external components.

Figure 53:
External Components - Capacitors

Symbol	Parameter	Temp. Characteristic	Min. Rated Voltage	Max. Tolerance	Max. ESR	Min./Max. Nominal Capacitance(1)	Recommended Typ. Component Value
C_{VANA}	Input capacitor AVDD LDO supply pin	Y5R; X5R	4 V	±10%	4 Ω@ 1 MHz	0.7 µF/2.2 µF	2.2 µF
C_{AVDD}	Output capacitor AVDD LDO regulator output	Y5R; X5R	4 V	±10%	4 Ω@ 1 MHz	0.7 µF/2.2 µF	2.2 µF
C_{VDDC}	Input capacitor of digital core supply LDO	Y5R; X5R	4 V	±10%	4 Ω@ 1 MHz	0.7 µF/2.2 µF	2.2 µF
C_{CVDD}	Output capacitor of digital core supply LDO	Y5R; X5R	4 V	±10%	4 Ω@ 1 MHz	0.7 µF/2.2 µF	2.2 µF

(1) This is the effective capacitor value considering component tolerances, voltage derating and temperature derating at selected operating voltage.

Below there are some explanations of the coupling capacitors of the AS3460 IC:

The VANA supply pin with an input supply range of 1.7 V up to 3.6 V is used to source the analog audio LDO, which provides the necessary supply voltage for the DAC and headphone amplifier. The integrated regulator allows AS3460 to be supplied directly with a DCDC converter maintaining highest audio quality level. This pin needs a DC input blocking capacitor of 2.2 µF.

The output clocking capacitor C_{AVDD} is needed to guaranty stable operation of the LDO over the complete operating range of AS3460 and should be 2.2 µF.

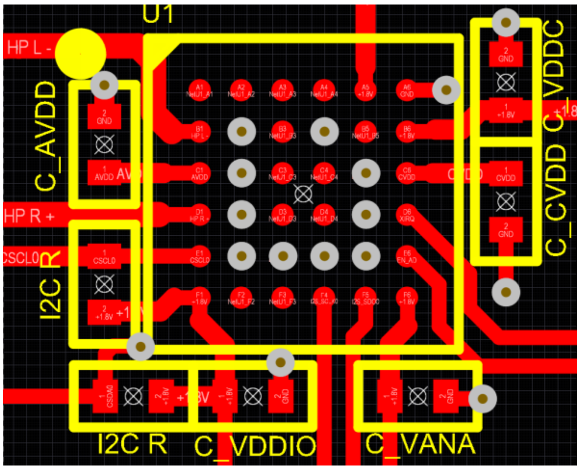
The VDDC pin sources the second LDO, which generates a 1.1 V output voltage to supply the MCU, AHE and PLL as well as the memory. This pin requires a DC input blocking capacitor C_{VDDC} and an output blocking capacitor C_{CVDD} to guaranty stable operation of the LDO over the complete operating range of AS3460. These pins need a 2.2 µF capacitor.

The highest system integration flexibility the GPIO interface PORT A and the chip control interface have two dedicated supply pins VDDPA and VDDIO.
The pin VDDIO needs a 100 nF capacitor and VDDPA does not need a capacitor; connect pin directly to 1V8.

Very important is that these capacitors (mentioned above) are in the layout as close as possible next to the AS3460 pins.

The layout example below highlights this again.

Figure 54:
PCB Layout Top Layer – Components On Single Side, Not All Pins Are Routed



The headphone amplifier are blocks with higher system currents, therefore use wide signal lines for headphone wires and for the supply. A weak signal line on any of these connections can influence channel separation of the device.

The ground pins of the AS3460 IC and the external components are directly connected through Vias to a dedicated ground (GND) plane below this area. The ground plane should be connected to the battery terminal for best grounding effect.

Figure 55:
PCB Layout GND Layer – Components On Single Side, Not All Pins Are Routed

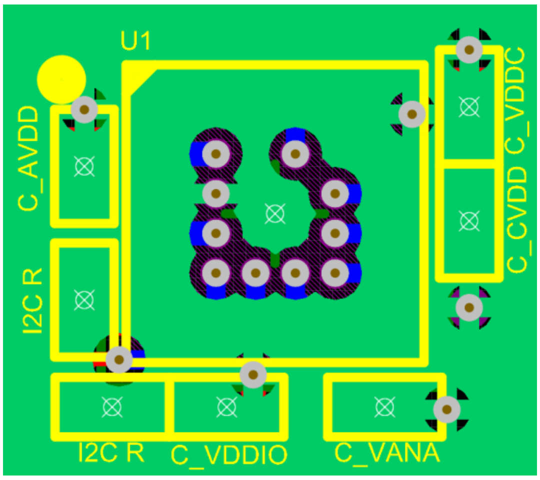
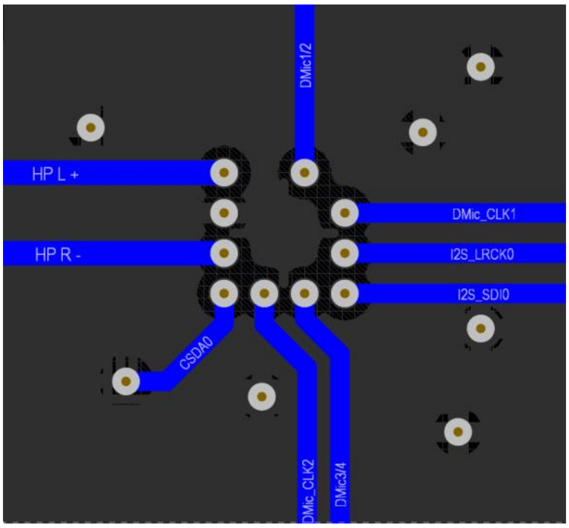
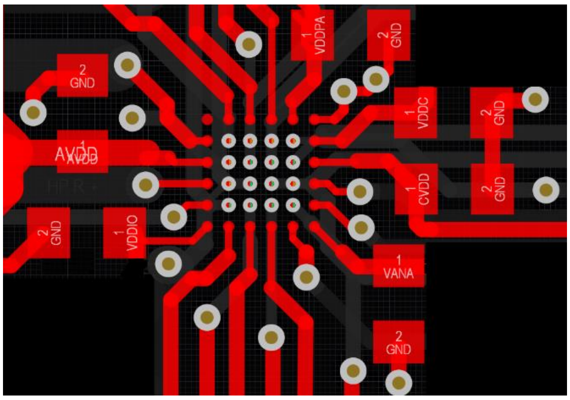


Figure 56:
PCB Layout Bottom Layer – Components On Single Side, Not All Pins Are Routed



If user has to route all pins out, there is an example shown below. As already mentioned above, it is important to put the coupling capacitors as close as possible to the supply pins.

Figure 57:
PCB Layout Top Layer – All Pins Routed



Ensure that the digital signal paths (digital microphones, I^2C etc.) are not below/ above or crossing the headphone path wires on other layers. Between this digital signals should be a GND layer (for shielding). The digital signal wires should be as short as possible to avoid irradiation.

Figure 58:
PCB Layout Signal Layer 1– All Pins Routed

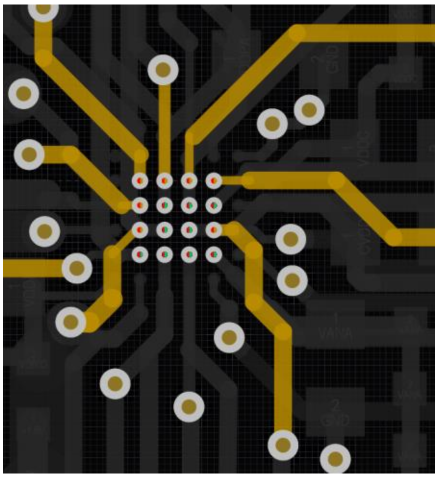


Figure 59:
PCB Layout Signal Layer 2 – All Pins Routed

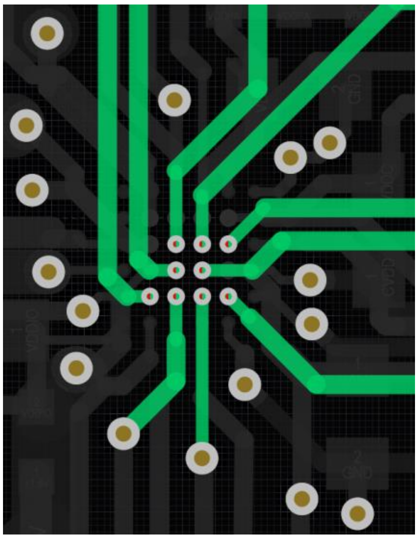
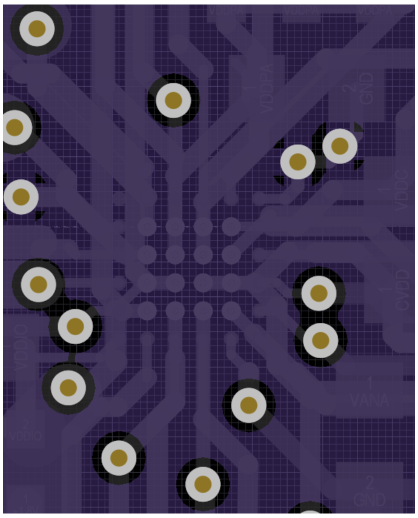
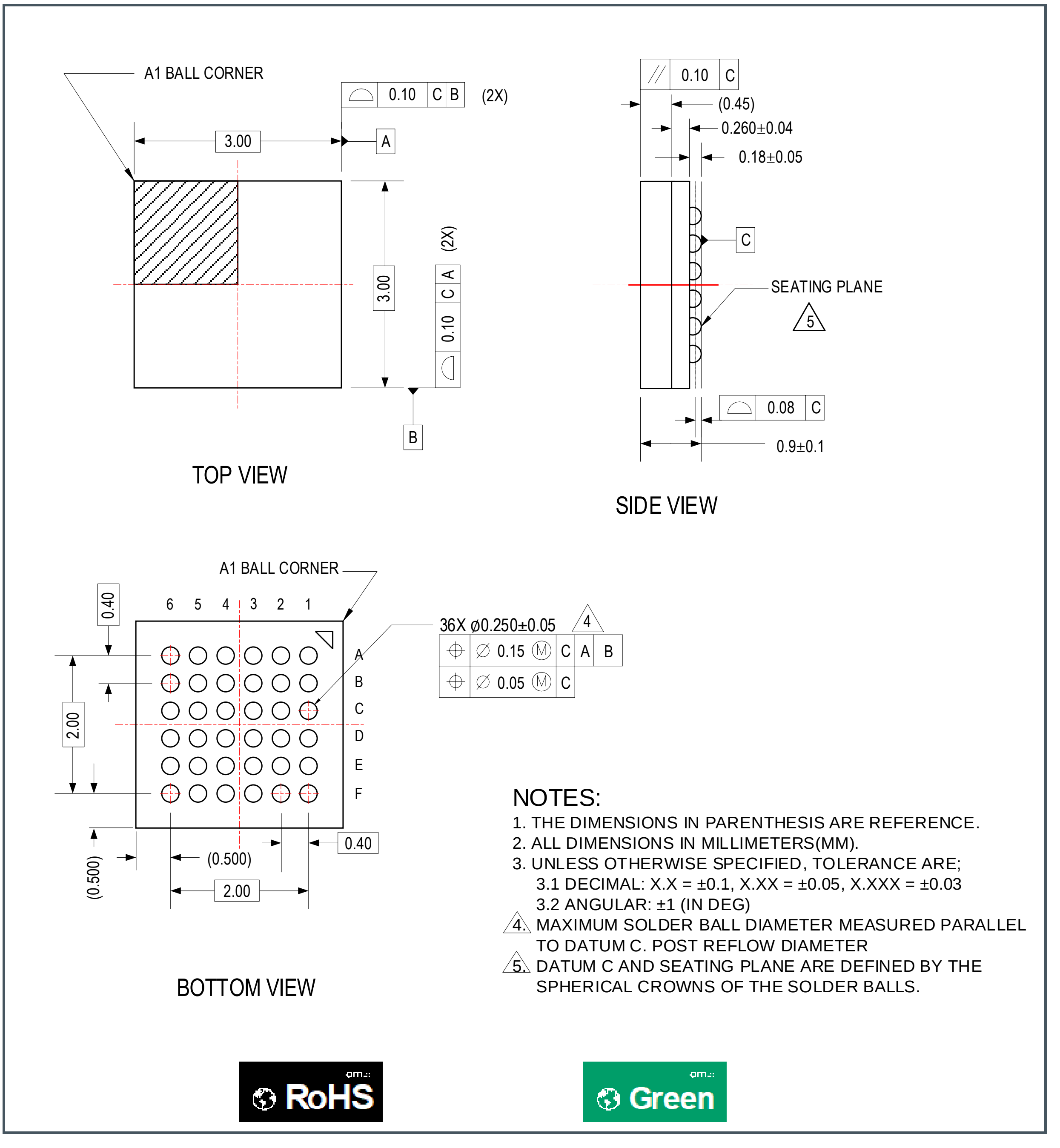


Figure 60:
PCB Layout GND Layer – All Pins Routed



4.1.2 Package Outline Drawing

Figure 61:
AS3460 Package Drawing (FBGA36)



5 FleX Filter Description

In this section the different filters which are available in FleX are described. Please find also a description of how to import biquad filter coefficients into FleX.

5.1 Filter Tabs

Next to the leftmost column the tuning tab is structured in subtabs, one for each filter. All subtabs have an identical structure, with one column for each filter. There are several filter types. Some filter stages are fixed in the system, while others stages can have different filter types and can be bypassed.

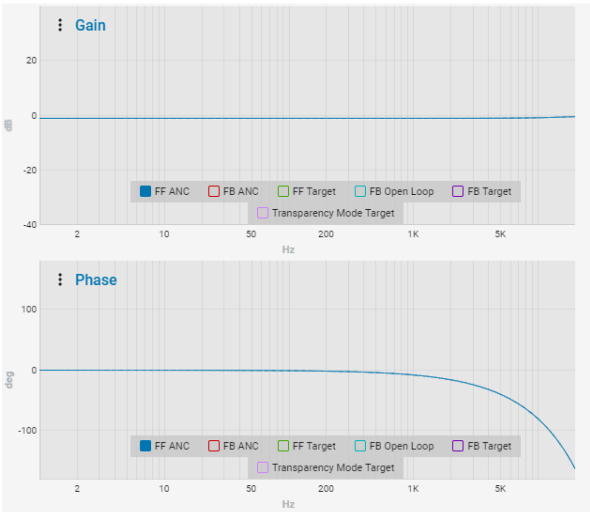
Filter Controls

The filters can be controlled by either dragging the control knobs, using the scroll wheel while the cursor is over the knob or manually entering a value into the box below each knob. By pressing CTRL the step size can be reduced, by pressing shift the step size can be increased. A double click on the knob resets it to its default value.

Downsampler

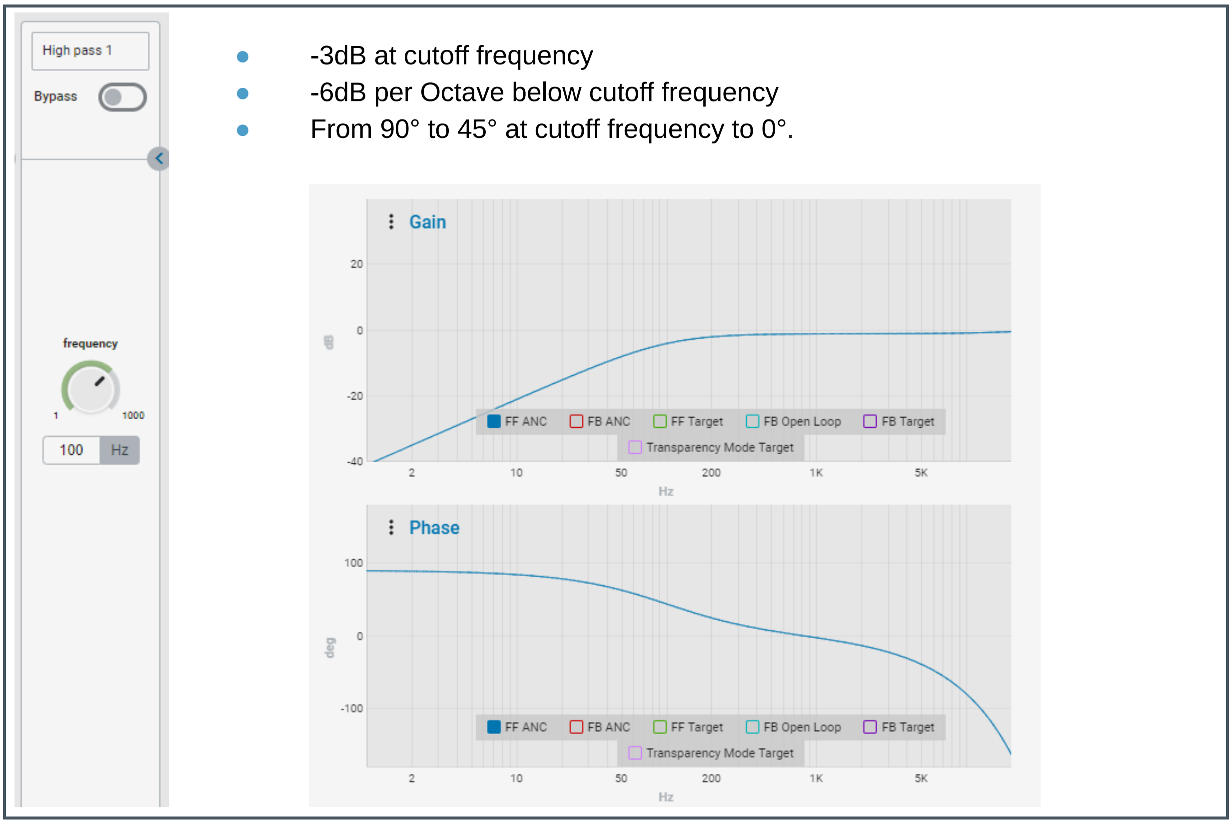
Figure 62:
Downsampler

- Default latency of 22 µs
- No impact on gain
- Lowers phase for high frequencies
- Reducing the latency below the default value raises the noise floor.



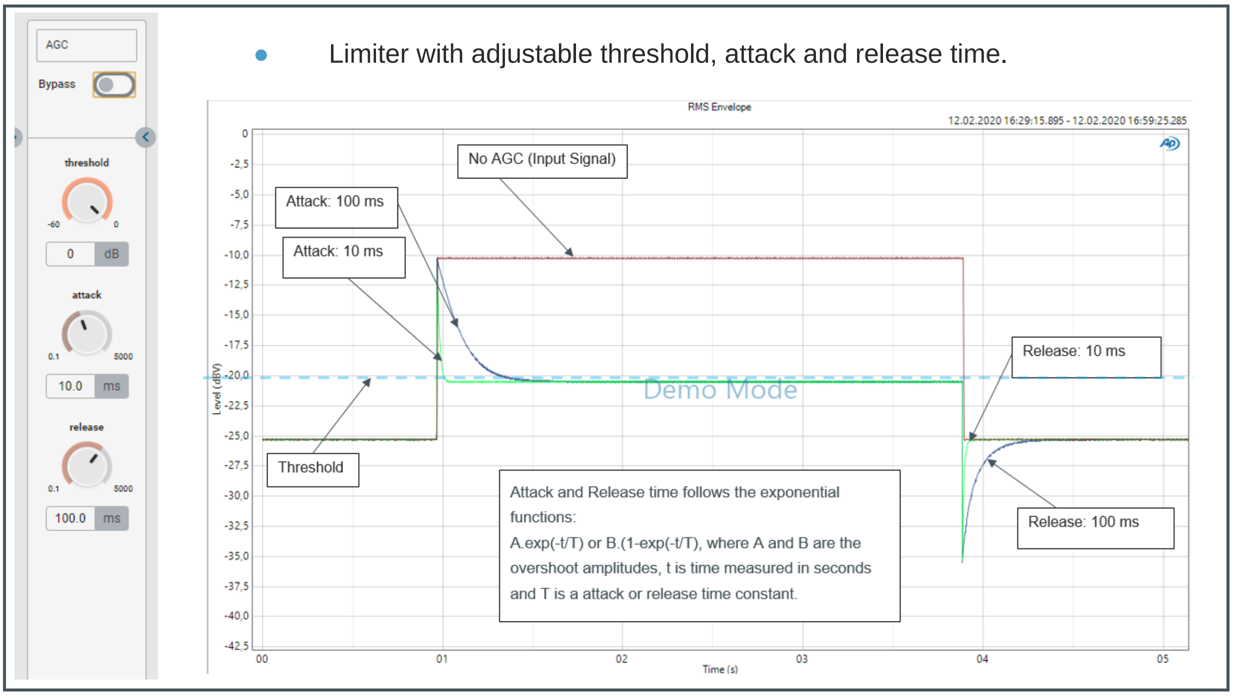
High Pass

Figure 63:
High Pass



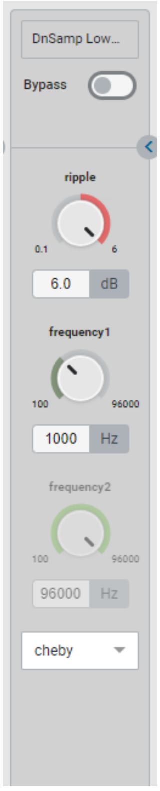
Automatic Gain Control (AGC)

Figure 64:
Automatic Gain Control

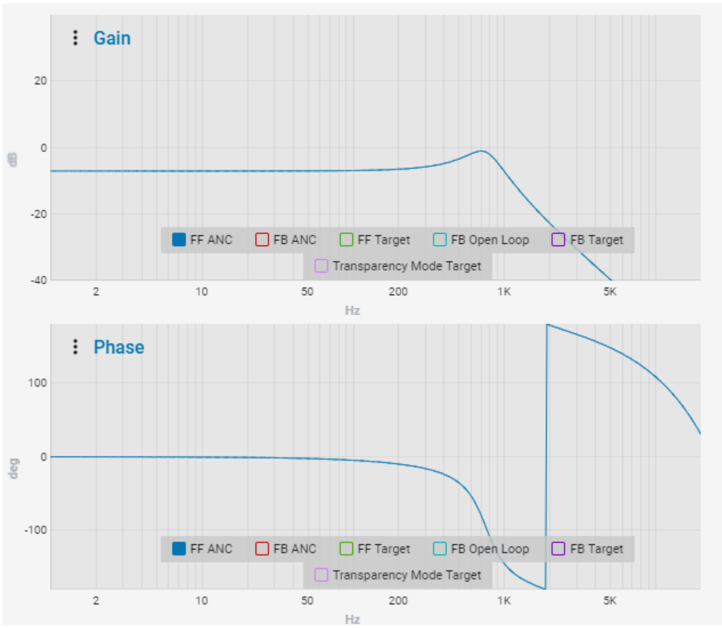


Downsampler Lowpass

Figure 65:
Downsampler Lowpass

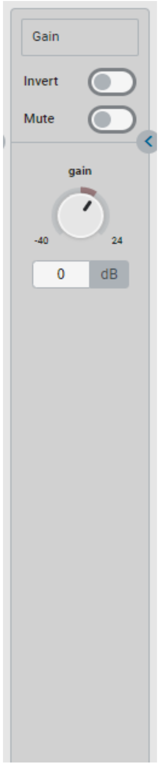


- Four filter types: LowpassX1, LowpassX2, Chebyshey, Butterworth.
- Available settings and frequency response depend on filter type.
- Below example of Chebyshev from left column.



Gain

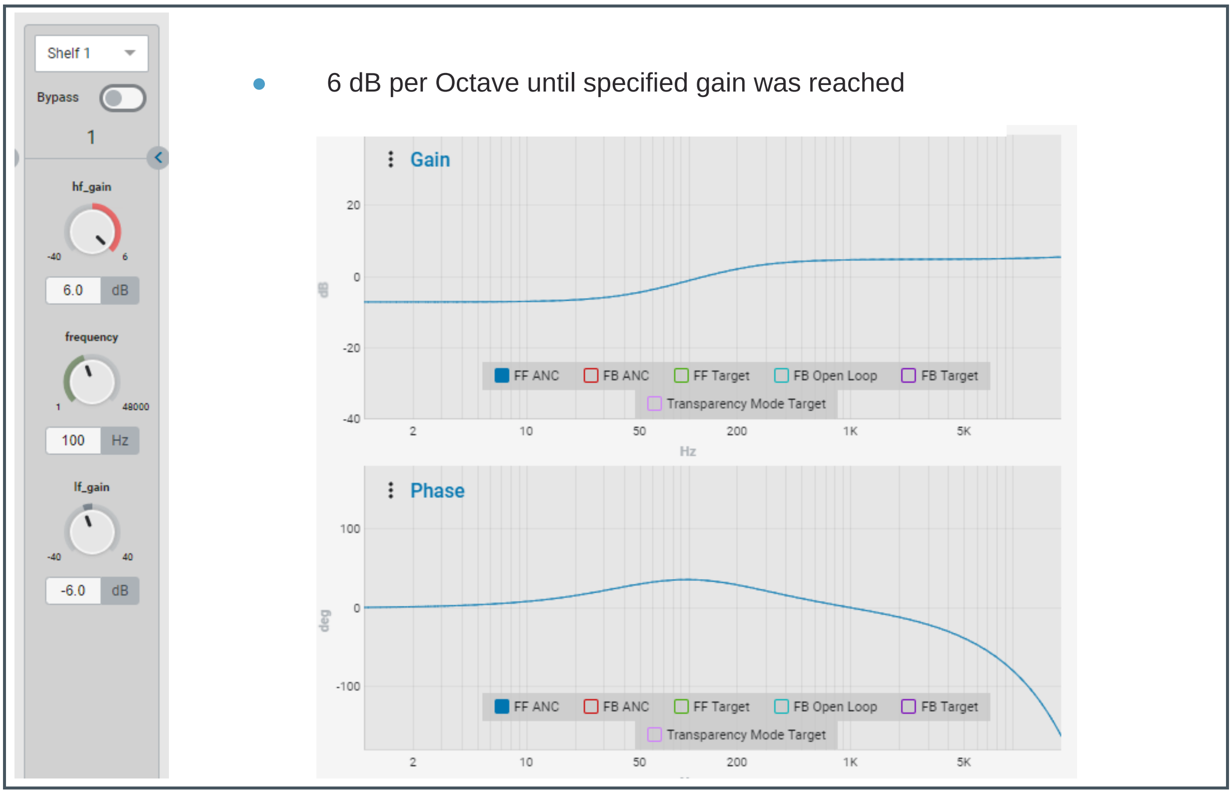
Figure 66:
Gain



- Gain of up to 24 dB.
- Inverting provides a phase shift of 180° without affecting the amplitude.

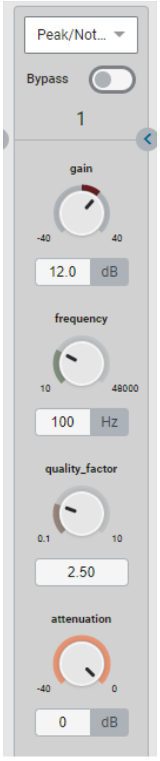
<u>Shelf</u>

**Figure 67:
Shelf**

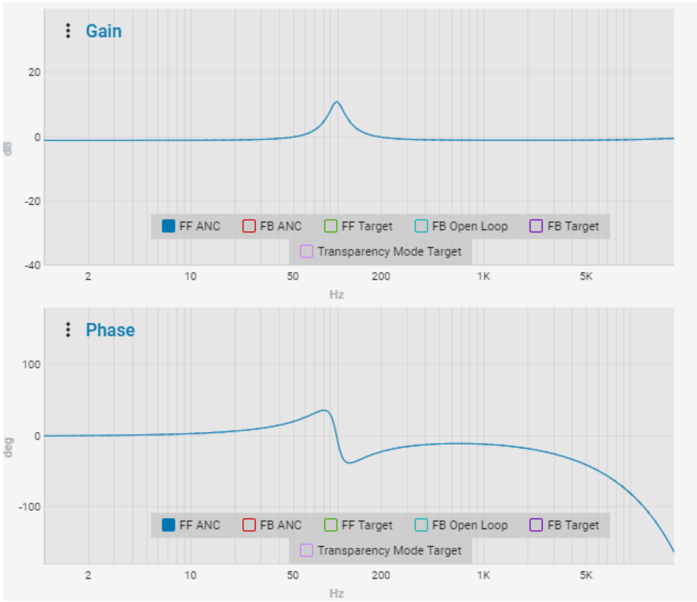


Peak or Notch

Figure 68:
Peak or Notch

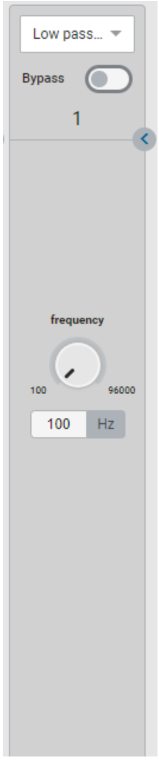


- Specified gain at center frequency.
- Adjustable Q-Factor or bandwidth.
- 0° phase shift at center frequency

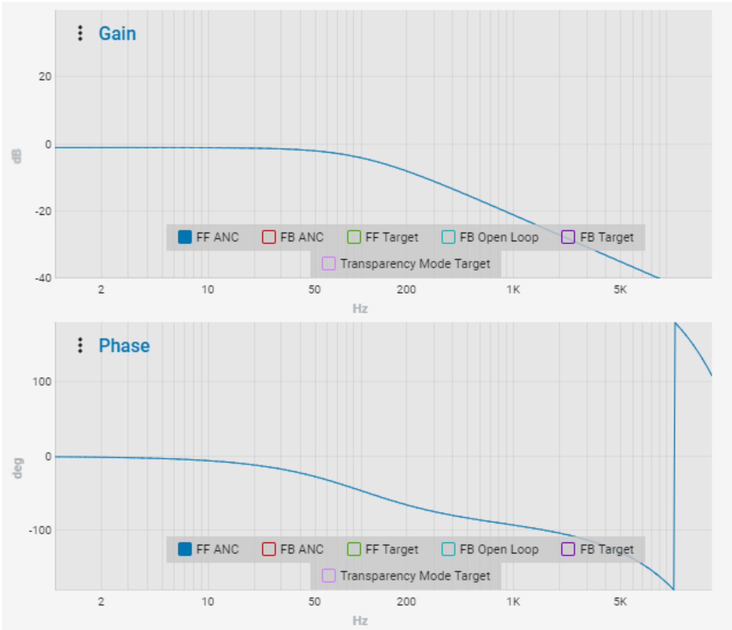


Low Pass

Figure 69:
Low Pass

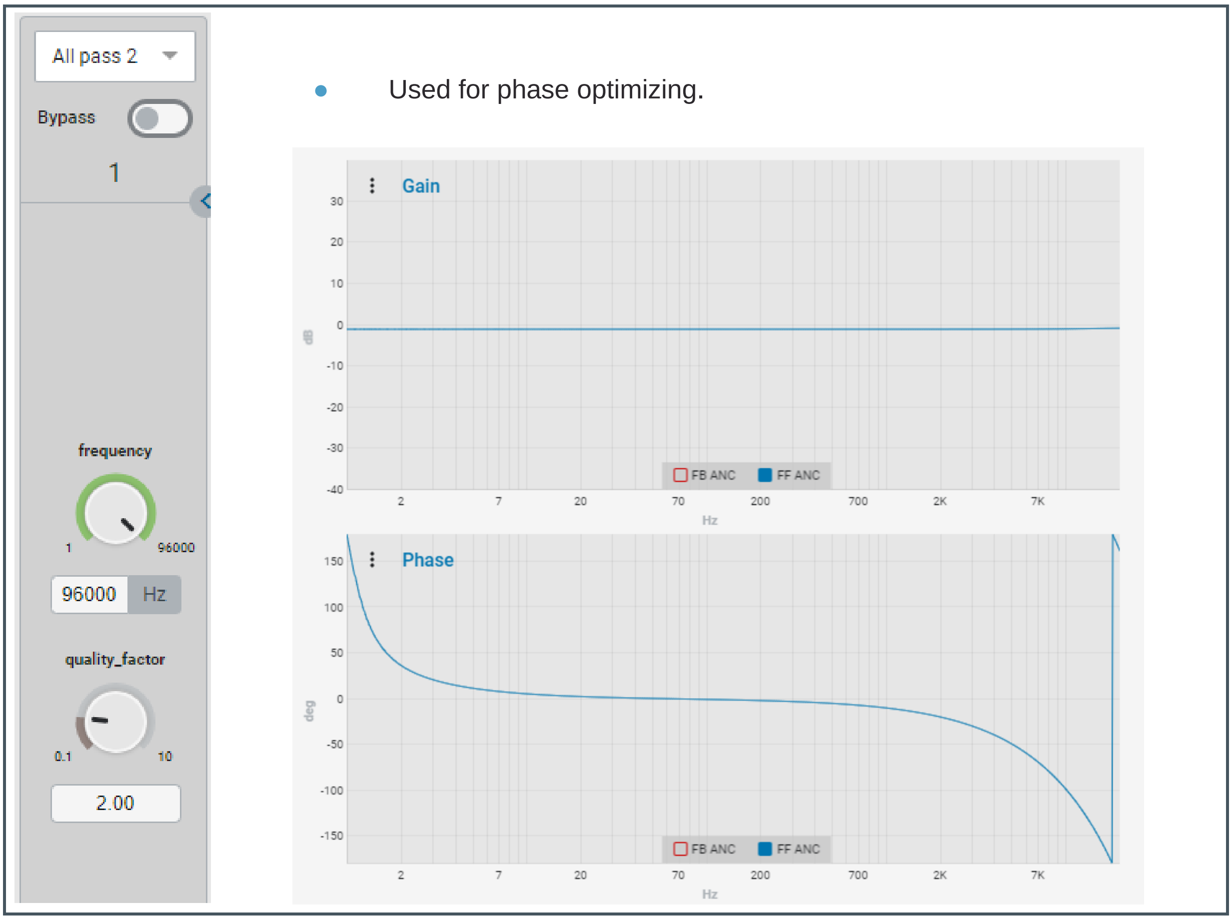


- -6 db per octave above cutoff frequency.
- 0° to -45° phase shift at cutoff frequency to -90° above.



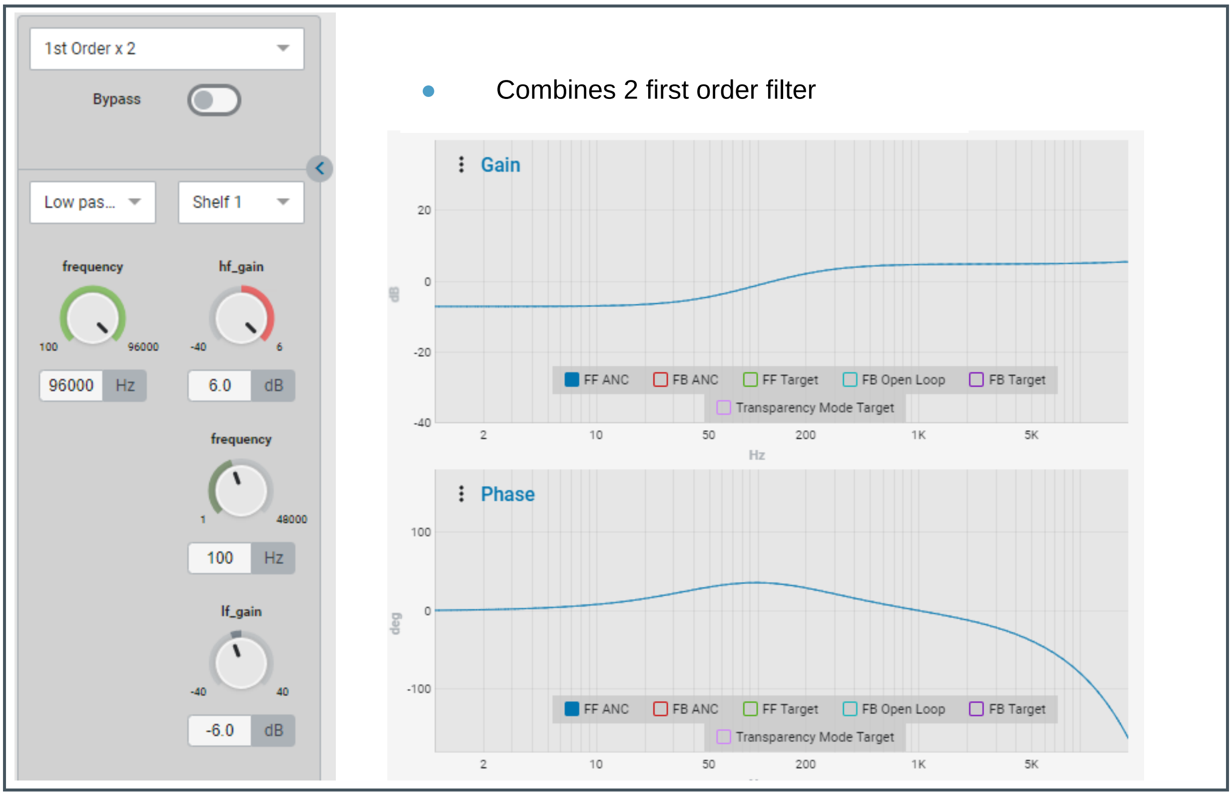
Allpass

Figure 70:
Allpass



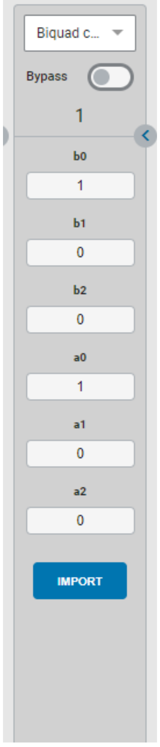
1st Order x2

Figure 71:
1st Order x2



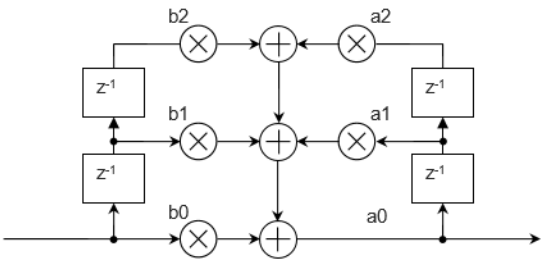
Biquad Coefficients

Figure 72:
Biquad Coefficients



- Option to import biquad coefficients calculated externally
- Follows the convention:

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{a_0 + a_1 z^{-1} + a_2 z^{-2}}$$



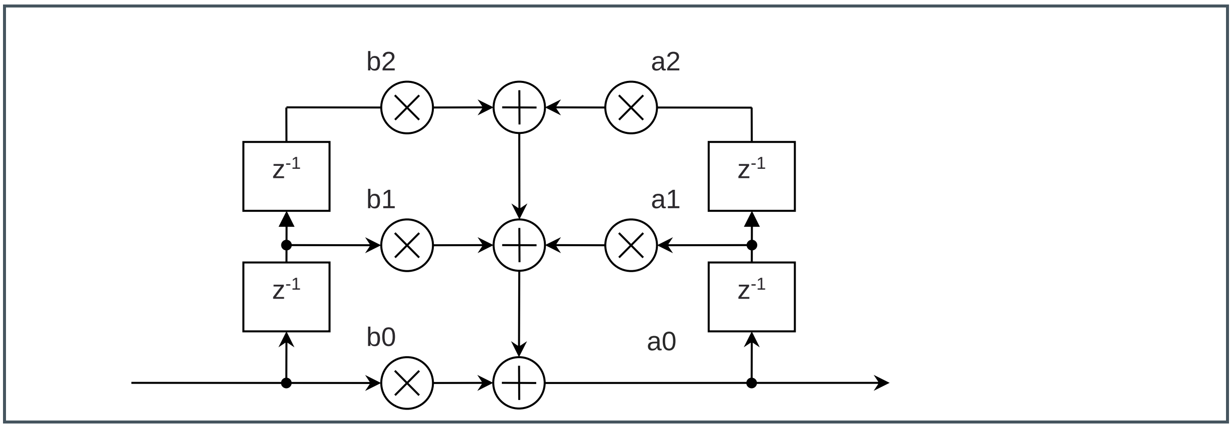
- a_0 is fixed to one, other coefficients have to be in the range [-2.0, 2.0).
- Coefficients can also be imported from ASCII files.
- There are certain rules regarding the format of coefficients
- More details can be found at the “Importing Biquad Coefficients” section

5.2 Importing Biquad Coefficients

The FleX filter design tool allows customers to design filters for the AS3460 ANC device.

External coefficients can be imported into FleX as biquad. The term biquad is equivalent to the term “second order section”. Each biquad has 6 floating point coefficients, one of which (a0) is normalized to 1.0.

Figure 73:
Biquad Filter Structure



The coefficients are described below:

Figure 74:
Coefficients

Coefficient[1]	Range[2]	Description
b0	[-2.0, 2.0)	The first feedforward coefficient
b1	[-2.0, 2.0)	The second feedforward coefficient
b2	[-2.0, 2.0)	The third feedforward coefficient
a0	1.0	A constant value always equal to 1.0
a1	[-2.0, 2.0)	The first feedback coefficient
a2	[-2.0, 2.0)	The second feedback coefficient

(1) The coefficients for a single biquad (second order section)
(2) The notation [min,max) is the standard notation for half-open interval, including values greater or equal to min and less than max.

Values outside the stated range will be rejected and an error will be generated during the import process.

It is also possible to load high order filters, which are decomposed into multiple biquads, at once.

5.2.1 Biquad Filter Strip

To select the interface where the biquad filter coefficients can be inserted, please select “Biquad coefficients” in the filter strip menu. Then the biquad coefficients input mask appears:

Figure 75:
Interface for Controlling a Single Filter Block

Coefficients can be inserted either manually or via importing an ASCII Text file (button “IMPORT”).

It is possible to import one or multiple filters via ASCII Text import. To import more than one biquad filter the same amount of biquad filters written in the text file have to be activated in the “Filter” tab.

Examples for importing multiple biquad filters are given in Section 3.

5.2.2 Data Format

Requirements for the data in the file containing biquad coefficients are as follows:

- The coefficients will be imported from an ASCII text file in Windows format
- Each line of the text file will contain one set of biquad coefficients
- Each biquad filter is defined by 6 floating point numbers in the order b0, b1, b1, a0, a1, a2
- An error will be reported if the floating-point numbers exceed the half-open range [-2.0, 2.0)
- The value a0 shall always equal 1.0. (an error will be reported if a0 is not equal to 1.0)
- The numbers can be written exponential format (“1.234e-5”) or fixed point format (“0.001234”)
- A space, tab or comma can be used as separator between the coefficients
- A bracket (square, curly or round) may start and end each line of numbers, but will be ignored
- Comments can be written after a “*”,”#” or ”/”

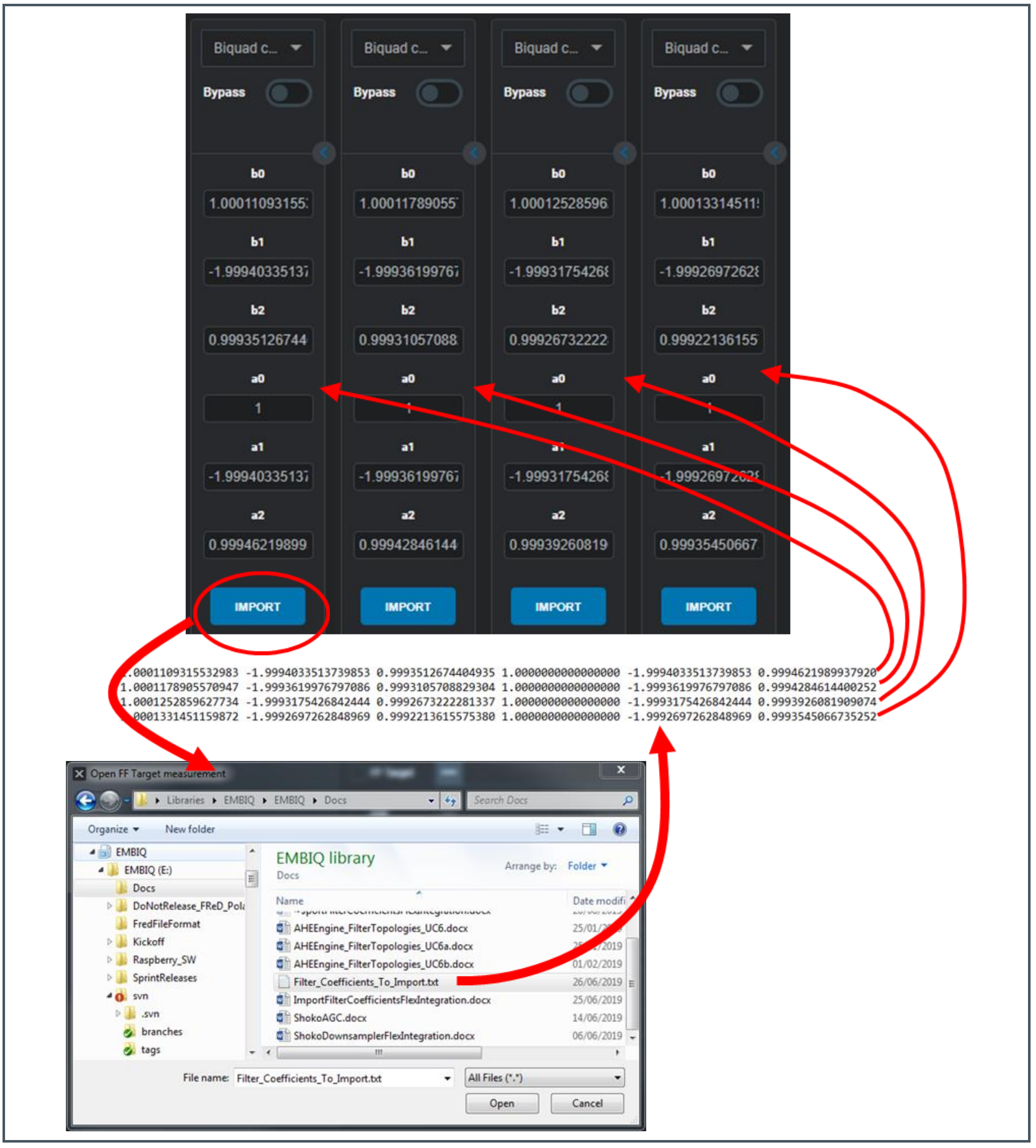
Examples using valid import formats are shown below:

```
[ 2.61652695e-04, 5.23305390e-04, 2.61652695e-04, 1.0 , -1.95372795e+00,
9.54774560e-01]

[ 0.02162072, 0.04324144, 0.02162072,  1. , -1.54312113, 0.629604 ]  # Low pass at
10kHz

0.99768867 -1.99537735 0.99768867 1. -1.99537201 0.99538269  * High pass at 100Hz

(0.50057783 -0.99882295 0.49826652 1.00000000 -1.9976459 0.99768869) / Peak at
200Hz
```

5.2.3 Examples

Multiple Biquad Import Example 1

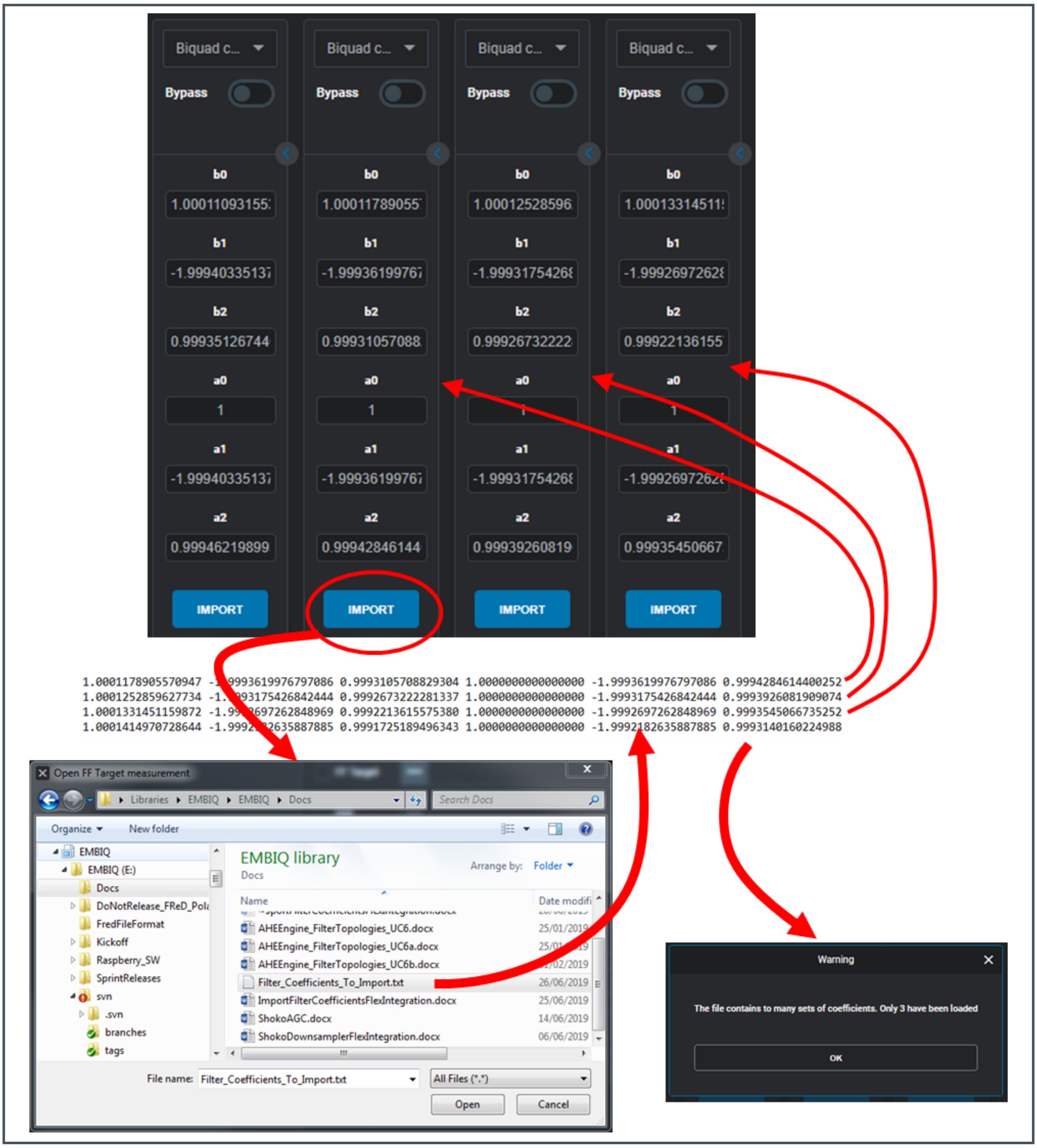
Figure 76:
Import Example 1



Import example where there are 4 "Biquad Coefficients" controls and 4 sets of coefficients in the imported file. The operation succeeds with no errors or warnings

Multiple Biquad Import Example 2

Figure 77:
Import Example 2



Import example where there are multiple “Biquad Coefficients” controls in a row, but there are more sets of coefficients in the imported file than there are controls.

6 ANC Tuning Guide

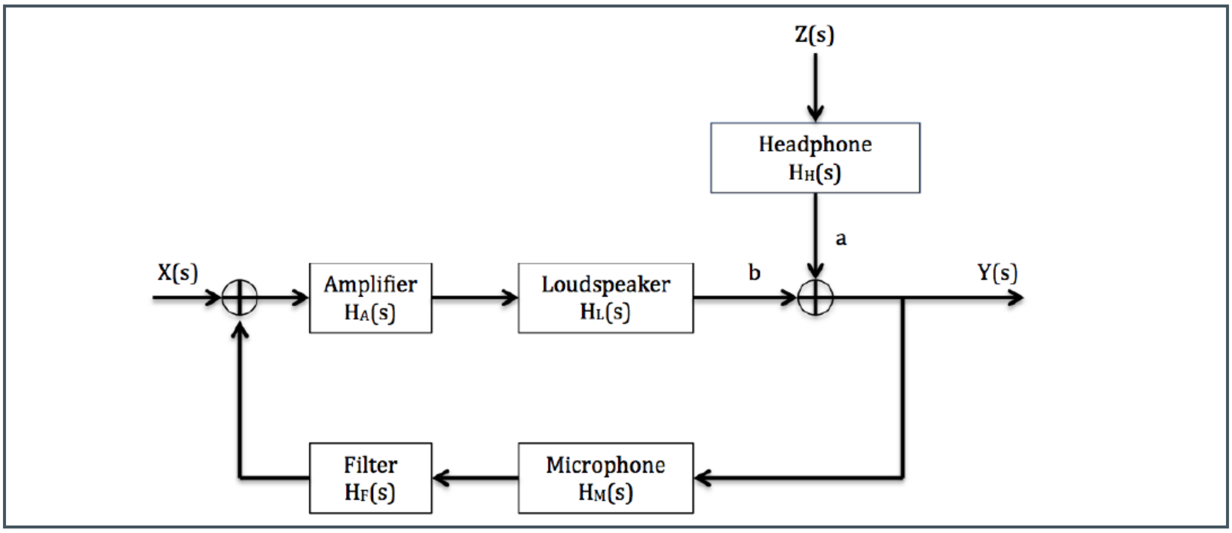
The following section will demonstrate how to tune a feedback system.

6.1 Feedback

6.1.1 Theory

In a feedback ANC system, the microphone is positioned inside the ear cup, next to the speaker. The microphone is sensing the noise coming from outside, already damped by the passive attenuation of the headphone cup. Also, the microphone senses the output of the speaker, building a feedback loop, therefore the system is a closed loop system.

Figure 78:
System Description



Using the system description in the Figure 78, the goal of a feedback ANC system can be stated as:

Equation 1:

$$Y(s) = 0$$

This leads (without derivation) to the ideal filter transfer function:

Equation 2:

$$\mathrm{H_F}(s) = -H_M(s)^{-1}H_A(s)^{-1}H_L(s)^{-1}$$

The three transfer functions on the right hand side are often grouped together as the “Driver to Feedback Microphone” transfer function:

Equation 3:

$$DFBM = H_M(s)H_A(s)H_L(s)$$

These can be obtained by a single measurement, described in the App Note on “Characterization”.

The ideal feedback filter would be the perfect inverse of the DFBM transfer function, but it is not possible to design such a filter. Only for a limited bandwidth, typically located between 20 Hz and 1 kHz it is possible to match the phase of the ideal filter close enough to achieve cancellation.

This leads to the goals when designing a feedback ANC-filter:

- Match the phase of the ideal filter as close as possible for as much bandwidth as possible
- Where the phase matches close enough, try to get as much filter gain as possible
- Where the phase does not match, try to reduce the gain as much as possible to ensure stability

Since the phase and gain of a filter can generally not be controlled separately, it is not a trivial task to find the optimal filter.

Additionally, the DFBM transfer function is subject to changes depending on the fit and user handling of the earphones and it should be guaranteed that the control loop remains stable in all use cases.

6.1.2 Tuning Guide

The tuning of the feedback path typically is an iterative process. One initial filter design will be developed and then tested for performance and stability. Since there is no simulation for all misuse cases, each filter needs to be hand tested for stability and if necessary adjusted until both ANC-performance and stability are satisfactory.

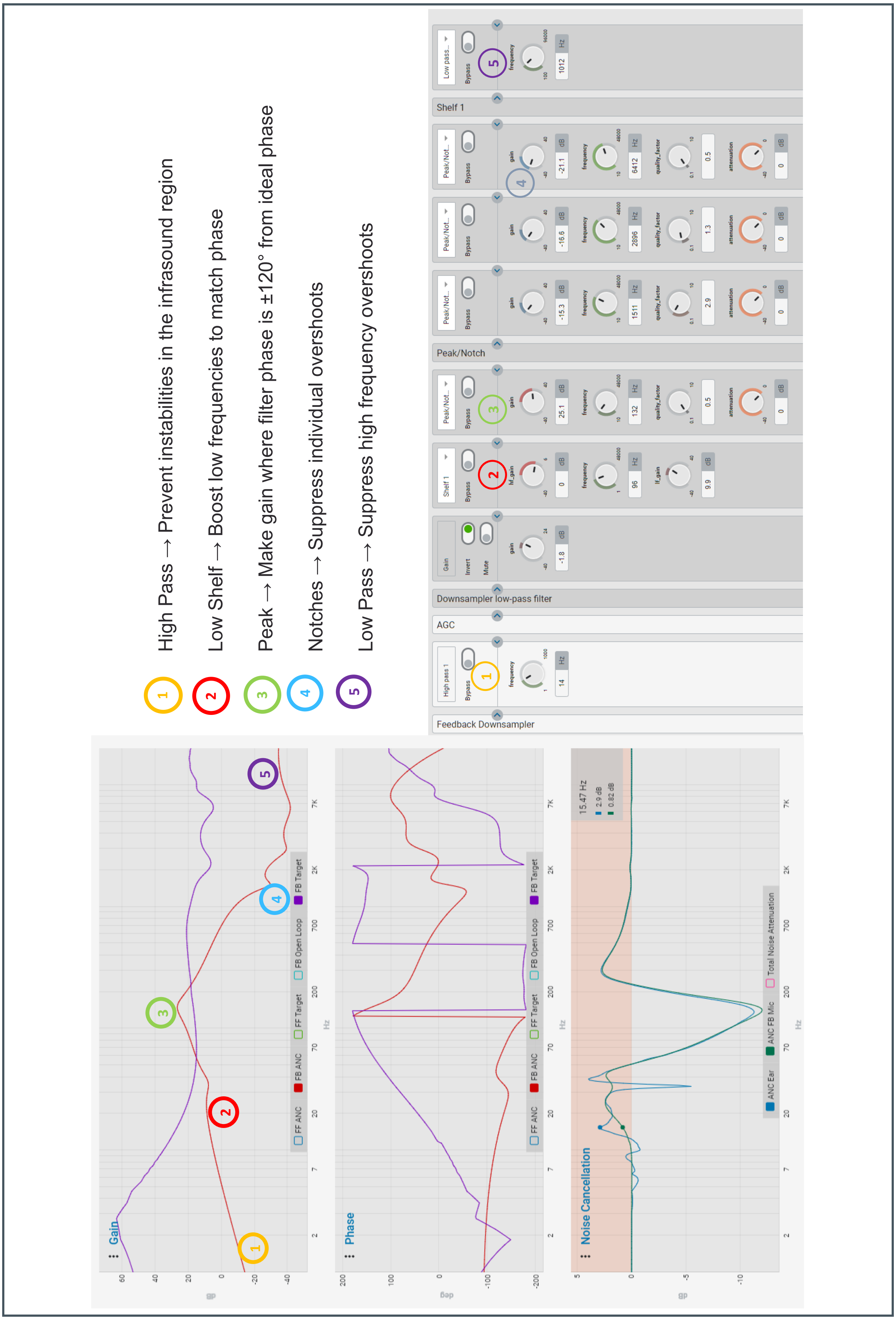
A typical design strategy would be:

- Bring down the filter phase close to the ideal phase at a frequency as low as possible, for example by boosting low frequencies using a shelf or a wide peak
- At the same time do not produce too much low frequency gain to ensure stability and little overshoot
- Overshoots at high frequencies can by suppressed using a low pass and notch filters at critical frequencies
- Performance bandwidth and strength can be extended by using peak filters

The optimal filter is a compromise between stability and performance, where the maximum achievable performance is a property of the headphone's acoustics. A large speaker will have a better low frequency response, allowing to bring down the phase at a lower frequency without requiring too much bass boost and less gain in the filter reduces stability problems.

6.1.3 Example

Figure 79:
ANC Tuning Example (see next page)


1 High Pass → Prevent instabilities in the infrasound region
2 Low Shelf → Boost low frequencies to match phase
3 Peak → Make gain where filter phase is ±120° from ideal phase
4 Notches → Suppress individual overshoots
5 Low Pass → Suppress high frequency overshoots

6.1.4 Performance Evaluation

Stability Testing

The next step after loading the new filter to the chip ("Write to AHE") is to test whether this filter is stable in all use cases. An end-user should never experience howling sounds in his ear or feel that the earphone is causing a pressure sensation (this would be due to low-frequency overshoots). Also the ANC-system should behave normal when confronted with high SPL sounds. The following list gives a hint of what tests can be conducted to evaluate the ANC-system's stability:

Figure 80:
Test to Evaluate ANC-System Stability

Stability Issue	Test
Too much low frequency gain	• Listen for pressure/pumping sensation • Listen with low frequency test sounds • Inside of driving car
Filter close to being unstable (might become unstable on a different sample)	• Cup speaker or microphone with hand • Close speaker opening • Press on speaker • Listen for resonant frequencies
High SPL sensitivity	• Slam a door next to the headphones • Playback music, press on the ear cup while wearing the headphones and listen for stuttering

Performance Test

ANC-Performance can be measured as described in the App-Note on ANC-Performance measuring, but it is also important to verify that the measured performance correlates with the simulated performance. If this is not the case it could be that the characterization measurements were not calibrated correctly, or that the headphone's fit to the measuring device is not reproducible exact enough.

Also it should be tested if the subjective performance is matching the measured performance. A good test environment is to play back pink noise on full range speakers.

6.2 Feed Forward

6.2.1 Theory

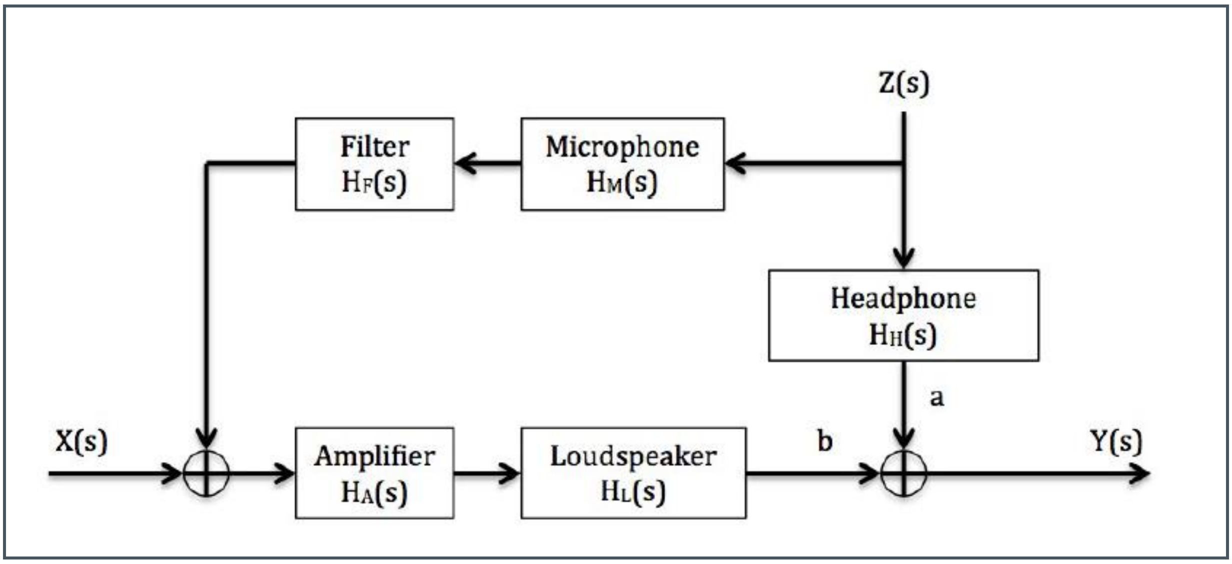
While feedback ANC is based on a control loop, the feed forward ANC is based on a different principle: As sound passes from the outside of the headphone through the headphone's structure to the ear, it is subject to a frequency dependent dampening. Quantifying this frequency dependent dampening gives the acoustic transfer function from outside to inside of the headphones.

If one would record the sound on the outside of the headphones, pass it through an exact replication of this transfer function and play it back inverted on the inside of the headphones, it would give perfect noise cancellation.

In the real world it is not possible to obtain a perfect replication, but it is possible to create an electrical or digital filter that can match magnitude and phase of the ideal transfer function to some degree over a certain bandwidth. Also the transfer functions of the speaker and the microphone have to be taken into account.

The following diagram and formulas illustrate the principle described above in mathematical terms:

Figure 81:
Feedback ANC Control Loop



Where $Z(s)$ is the noise and $Y(s)$ is the signal at the ear. $H_H(s)$ is the acoustic transfer function or passive response of the headphone. Now the aim is to get the combined transfer function of $H_M(s) \cdot H_F(s) \cdot H_A(s) \cdot H_L(s)$ to be as close a match to $-H_H(s)$ as possible. From the combined transfer function only the filter $H_F(s)$ can be changed. (Without derivation) We can calculate the ideal transfer function or **target** for the filter:

Equation 4:

$$H_F(s) = -\frac{H_H(s)}{H_M(s) \cdot H_A(s) \cdot H_L(s)}$$

It is not necessary to measure these transfer functions separately, as they are all included in the characterization measurement, yielding the equivalent expression:

Equation 5:

$$H_F(s) = \frac{H_{A2E}(s)}{H_{A2FF}(s) \cdot H_{D2E}(s)}$$

Where D2E is the “Driver to Ear”, A2E is the “Ambient Noise to Ear” and A2FF is the “Ambient Noise to Feed Forward Microphone” transfer function. This measurement procedure is described in detail in the App Note on “Characterization”.

The goals when designing the feed-forward filter are simple:

- Match the target both in phase and magnitude over a frequency range as large as possible
- Where a match is not possible, reduce the gain

6.2.2 Tuning Guide

Before starting the filter tuning it is important to evaluate the target curve for several samples and fits, to know what parts of the transfer function a reproducible and should be taken into account for the filter design and what parts change for every sample or due to the fit of the headphone. This way overfitting can be avoided and the filter will work on all samples with a similar performance.

To match the ideal filter or target curve it would usually be necessary to have an extremely high low frequency gain. The first step when starting a feed-forward filter design is to introduce a low shelf and sometimes also a wide peak at low frequencies, to identify the frequency at which the phase can be matched without producing too much low-frequency overshoot. The next step is to use peak and notch filters to follow the target curves. Finally, a low pass should be used to suppress the higher frequencies where no phase match is possible.

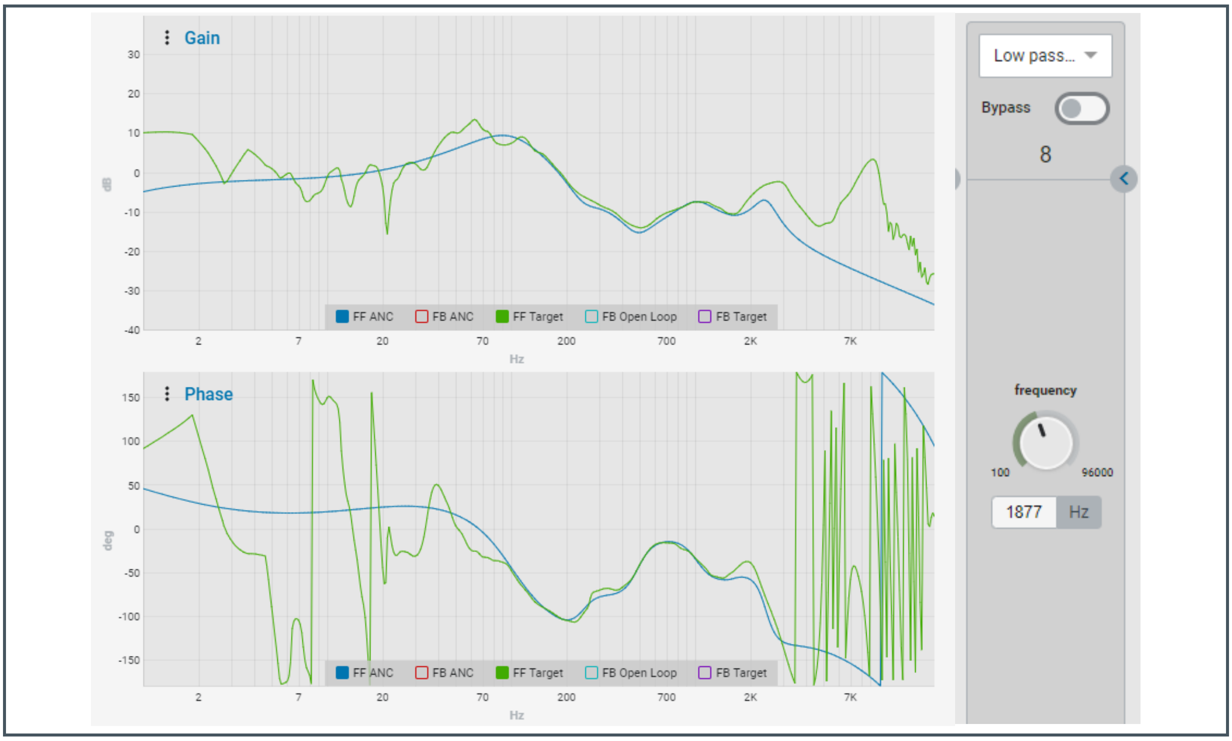
6.3 Example

The figures below document such a filter design procedure:

**Figure 82:
Example (part 1)**

Figure 83:
Example (part 2)

Figure 84:
Example (part 3)



7 ALC Tuning Guide

7.1 Introduction

The **ams** adaptive noise cancellation algorithm Active Leakage Compensation (**ALC**) is more complex than static ANC algorithms and requires the tuning of several filters and parameters. Luckily, there is no DSP programming or scripting required, because everything can be configured graphically within the **ams** FleX tool. The FleX tool provides a graphical interface for filter tuning, simulation, parameter setting and everything else there is to configure on AS3460. This tutorial shows the recommended design flow for configuring AS3460's ALC algorithm for a new earphone model.

Note: Only some concepts of operating FleX and the AS3460 evaluation kit are introduced here. Please also read the FleX manual and the evaluation kit handbook.

7.2 Overview

The whole, complex ALC algorithm can be split in several sections and sub-algorithms: there is the basic feed forward ANC adaption, a feedback ANC adaption derived from the feed forward adaption, the music compensation and adaption in the presence of music as well as several support algorithms for monitoring stability and other corner cases.

To handle this complexity, we recommend to follow a certain order when tuning the ALC algorithm:

1. Careful review and calibration of measurement data
2. General system configuration
3. Static tuning of feed forward ANC for the lowest leakage
4. Tuning of high leakage feed forward ANC
5. Feed forward only adaption and stability testing
6. For some headphones: Extra low frequency shelf
7. Design of feedback filters
8. Feedback filter performance and stability testing
9. Tuning of music compensation filters
10. Music compensation verification procedure
11. Testing and tuning of corner cases and algorithm parameters

7.3 Measuring ALC Headphones

One characterization measurement is not sufficient for ALC headphones; they must be characterized at five evenly spaced leakage conditions. The following section provides some information regarding how to perform these measurements and what to watch out for.

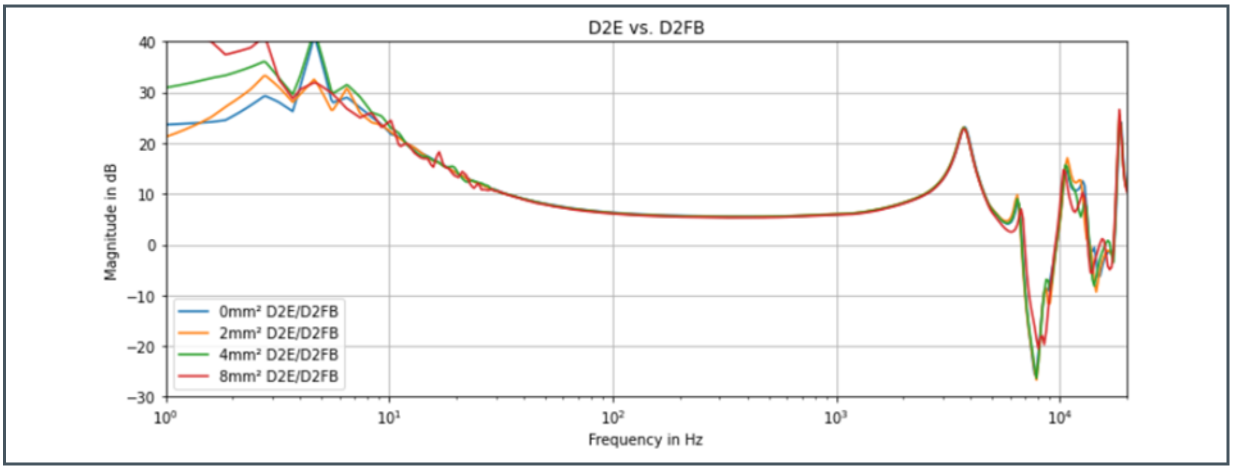
Note: The application note "Digital Headphone Characterization" describes the general procedure, setup and equipment for performing these measurements.

7.3.1 Characterization in Ear Simulator

The first series of measurements is used to check the acoustic properties of the headphone. It requires the use of an ear simulator, which can simulate variable leakage. We recommend the **ams** Leakage Adapter. Perform several characterization measurements starting at the 0 mm² leakage up to 28 mm² leakage for loose-fit or 8 mm² for sealed fit headphones.

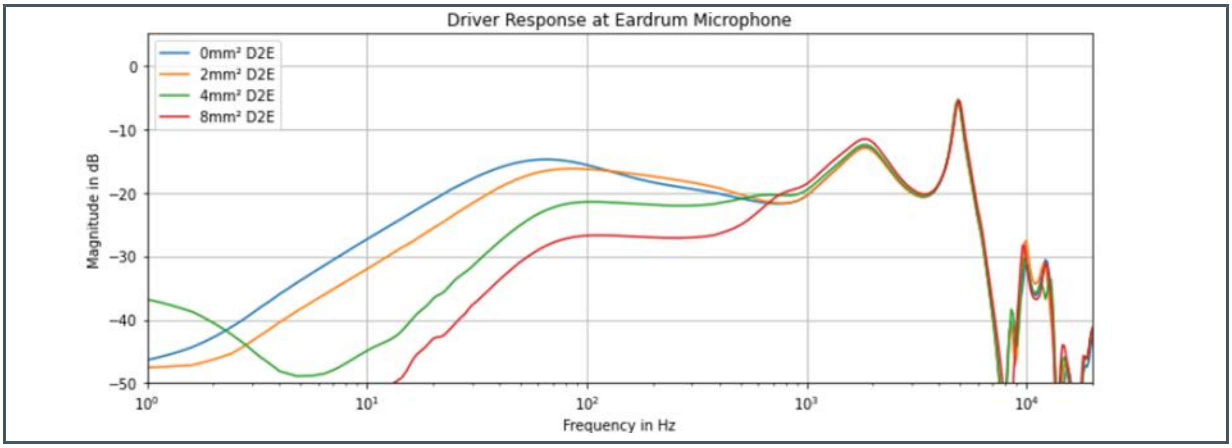
Examine these measurements regarding the isolation between speaker and feedback microphone. The gain difference of the driver response at the drum reference point microphone (**D2E**) and the driver response at the feedback microphone (**D2FB**) should be roughly constant between 50 Hz and 2 kHz and should not change with leakage. This confirms that the feedback microphone location requirement mentioned in both ALC and S-ALC Acoustic Design Guidelines is fulfilled. If this requirement is not met, the optimum that the algorithm adapts to, based on its error signal (the feedback microphone signal), is not the optimum at the eardrum. This means the user would experience noise cancellation, which is always off from the best possible performance. Figure 85 shows an example of a headphone with good feedback microphone to driver separation.

Figure 85:
Gain Difference of Driver Response at Ear to Driver Response at Feedback Microphone



By these measurements you can also confirm the functioning of the bass tube, see Figure 86

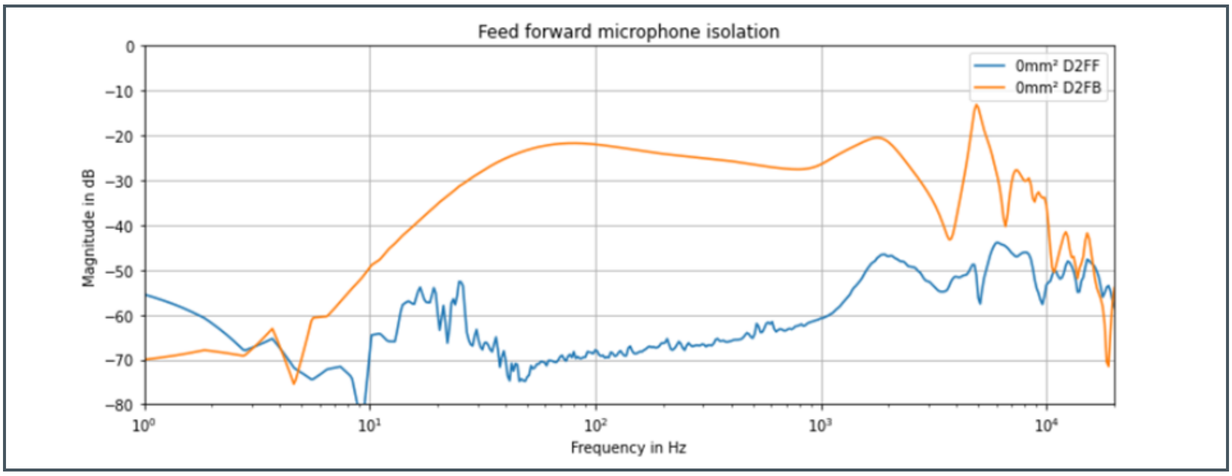
Figure 86:
Driver Response of ams Loose Fit Reference Design(1)



(1) Visible is the 'knee' at 80Hz, which is the basstube resonance. The basstube helps to maintain a good low frequency response at high leakages.

In addition, you can check the isolation of the feed forward microphone from the speaker:

Figure 87:
Driver Response at Feed Forward Microphone vs. Driver Response at Feedback Microphone(1)



(1) Too much driver signal reaching the feed forward microphone can cause instabilities by creating a parasitic feedback loop

7.3.2 Characterization in Human Ear

Determining the Leakage Range

The second series of measurements requires a number of human subjects. The purpose of these measurements is to determine the leakage range that occurs on real human ears. We do not recommend using an in-ear microphone, since even a thin wire distorts the fit and leakage and, since the correct feedback microphone placement has already been checked, the signal at the feedback microphone will match the signal of an in-ear microphone. It is sufficient to only record the feedback and feed forward microphone signals.

Low Leakage Test

Ask several people to insert the headphone at the lowest comfortable leak and characterize in this state. If the lowest leakage measured on an ear simulator is close to the lowest leak on most people's ear, the fit is good. If most people cannot fit the earphone in their ear at low leakage, the fit is bad, and they will not experience good ANC. Then the earphone's shape should be reconsidered. The amount of leakage between measurements can be determined by comparing the driver response at the feedback microphone.

High Leakage Test

Ask several people to insert the headphone at the highest comfortable leak, i.e. it should sit in the ear quite loosely, but still secure enough to not fall out and thereby representing the highest realistic leakage at which a user might wear the earphone. The algorithm works best if the driver response difference between the lowest and highest leakage is less than 20 dB across the ANC bandwidth (20 Hz to 3 kHz). If the difference is bigger, the adaption time will be slower and instabilities due to very high feed forward gains can occur. In this case, either reconsider the earphone's shape to provide a better fit or change the acoustic design (addition of a bass tube) to reduce the necessary bass boost in the feed forward filter. Alternatively, the highest leakage condition covered by the algorithm can be lowered. This way algorithm will work well, but will not provide good ANC for people on which the earphones fit too loosely.

7.3.3 Characterizing for Filter Design

Finally, the third series of measurements will produce the characterization data on which the filter design will be based.

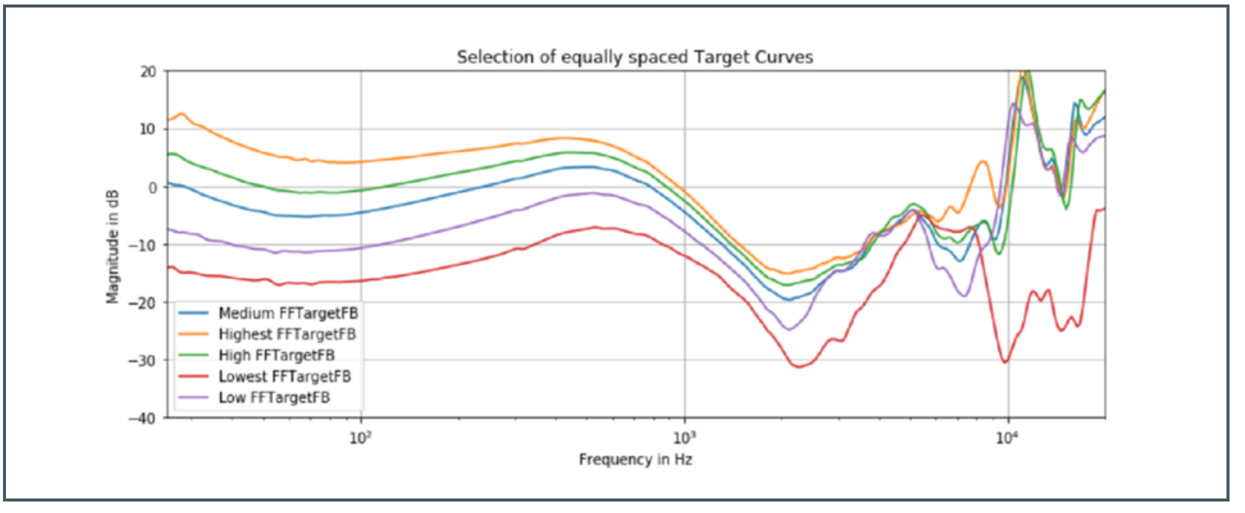
Using ams Leakage Adapter

We have found that the currently available ear simulators do not behave like real ears for higher leakages, which is why we have developed the **ams** leakage adapter. If you have access to an **ams** leakage adapter, you can re-use the measurements of the first series and select five characterizations based on equally spaced feed forward targets. The highest leakage is determined by matching to what was found to be the highest leakage on the human test ears.

Leakage Range

If you do not have an **ams** leakage adapter available, the filter tuning should be based on real ear measurements instead. Perform characterization measurements for three steps in between the lowest and highest leakage, measuring five leakage conditions. Ideally, these five steps are equally spaced. Figure 88 shows a selection of five equally spaced feed forward target curves with a range of approximately 20 dB. Since it is difficult for the test subject to produce equally spaced leakage conditions, we recommend to perform a number (>10) of characterizations and afterwards select those three target curves which cover the range between the highest and the lowest leakage with equal spacing. Although the filter tuning in FleX is based on tuning only two filter sets, one for the lowest and one for the highest leakage case, it is necessary to have measurements for three in between steps for the automatic calculation of algorithm parameters. Figure 88 shows an example of such a curve selection.

Figure 88:
Equally Spaced Target Curves



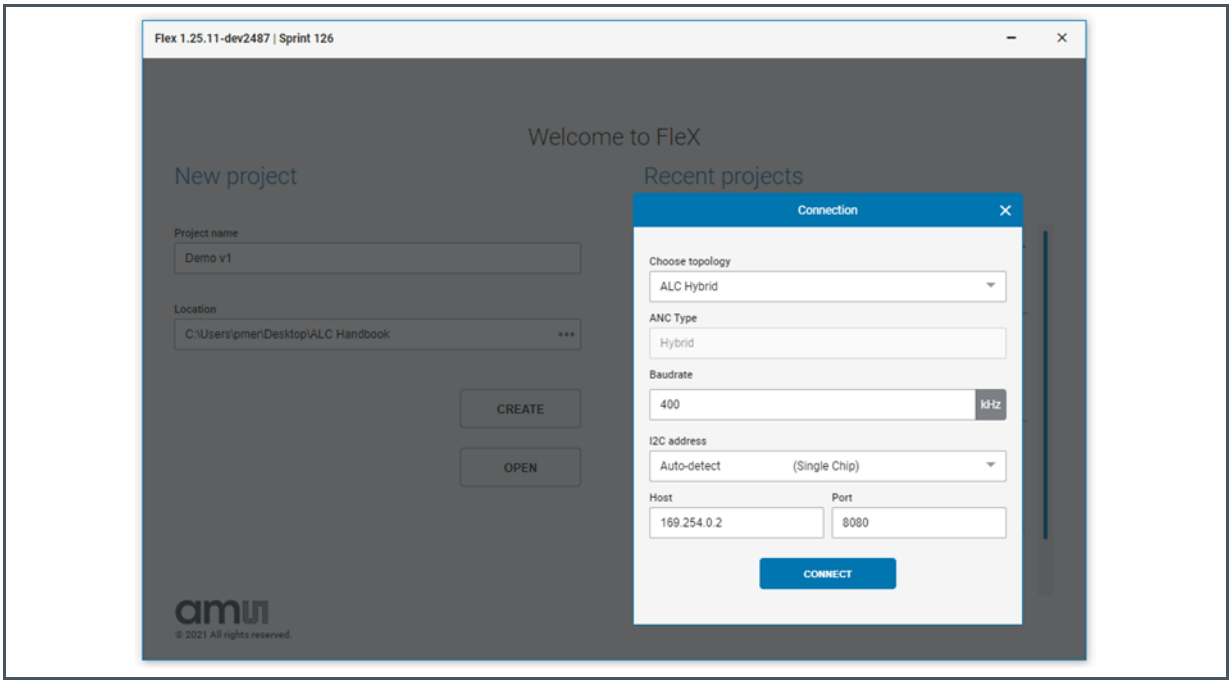
Note: Only use narrowband smoothing (e.g. 1/24th octave smoothing) as a too wide band smoothing can offset the target phase and produce an overpromising ANC simulation.

7.4 Project Setup

To create a new project, connect the **ams** AS3460 evaluation board (**EVK**) to your computer and launch FleX. Create a new project and choose "ALC Hybrid" as the topology. Then press connect. If the connection fails:

- The evaluation board's Raspberry Pi might not be finished booting yet, try again after a minute
- The two jumpers connecting the Raspberry Pi's I^{2}C and AS3460 could be missing (see AS3460 EVK)

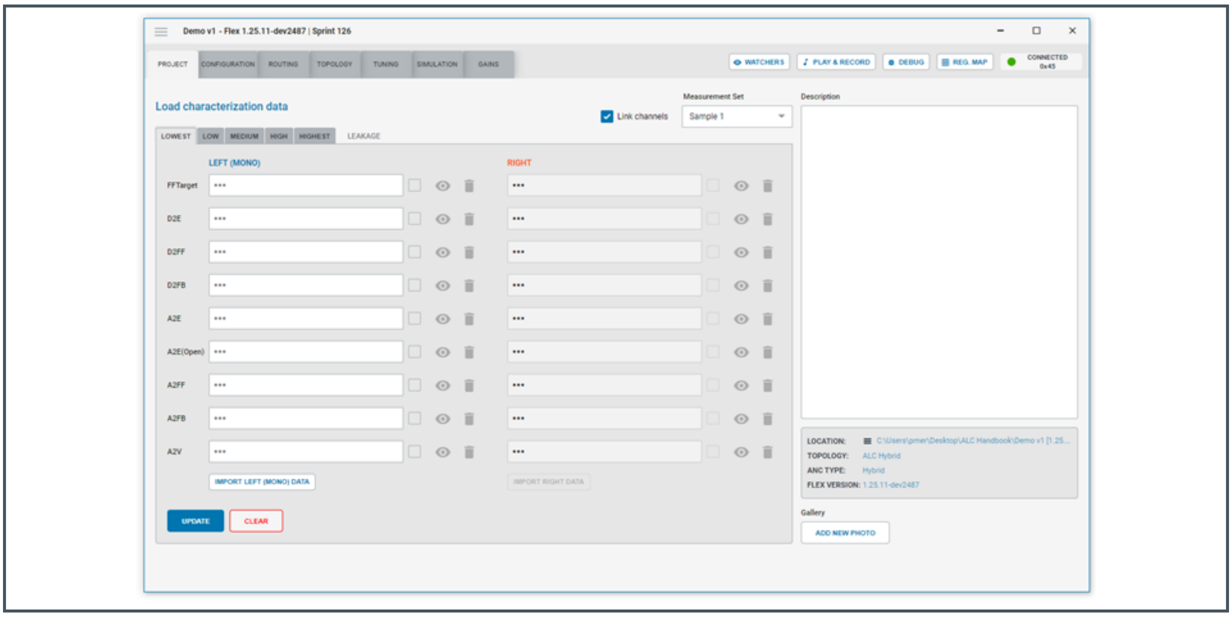
Figure 89:
FleX Connection Window



After connecting, you might be asked to update the Raspberry Pi backend and/or the AS3460 firmware. Confirm both, because a specific version of FleX will only be compatible with the correct backend and firmware. All versions of FleX are capable of up- or downgrading the firmware and backend as required.

Once updates are complete and the connection succeeded, FleX will open and initially display the Project tab. The project tab is used to load and calibrate the characterization measurement data for five leakages and different samples or subjects.

Figure 90:
FleX Project Setup



7.4.1 Importing Characterization Data

Note: General information on loading characterization data and its format can be found in the FleX handbook

Ear measurements (D2E, A2E…) are not required for ALC and are not available if your characterization was based on in-ear measurements. If you have used the **ams** Leakage Adapter and therefore have the D2E, A2E and A2E(open) measurements available, you can still import them, since they are useful for displaying the total noise attenuation and music equalization. However, all required calculations and simulations are based on the feed forward and feedback microphone responses: A2FF, A2FB and D2FF, D2FB. This works, because when following the ALC acoustic guidelines, the signal at the feedback microphone is equivalent to the signal at the eardrum over the frequency range of interest.

If the measurements are organized like in Figure 91, you can simply click "Import Left Data" and select the lowest leakage folder. Repeat this step for the other four leakages. If the measurement data does not follow this naming convention and folder structure, you have to manually select every measurement by clicking […] in the line for each measurement. After importing, click "Update". The target curve should now appear in the simulation tab.

Figure 91:
Recommended Folder and File Structure

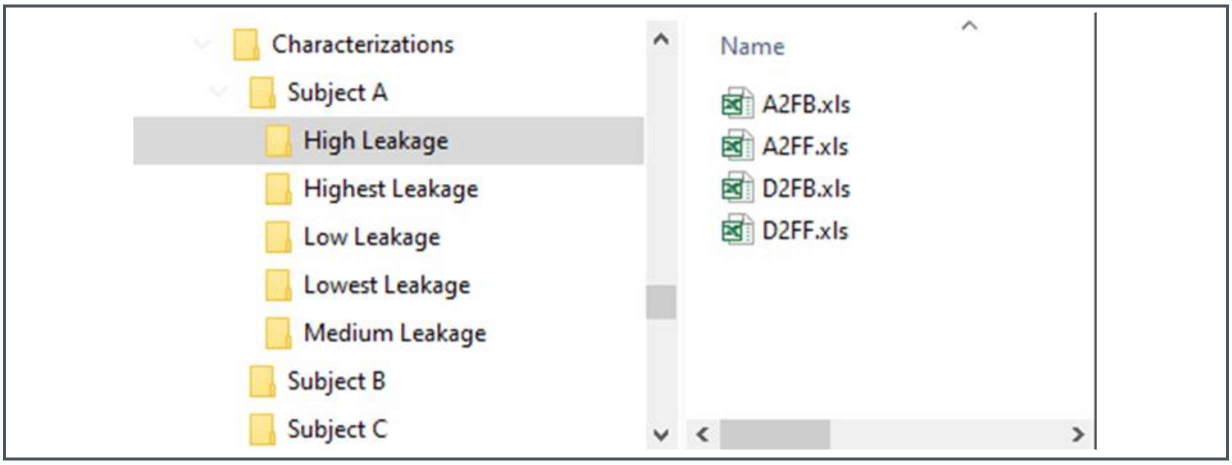
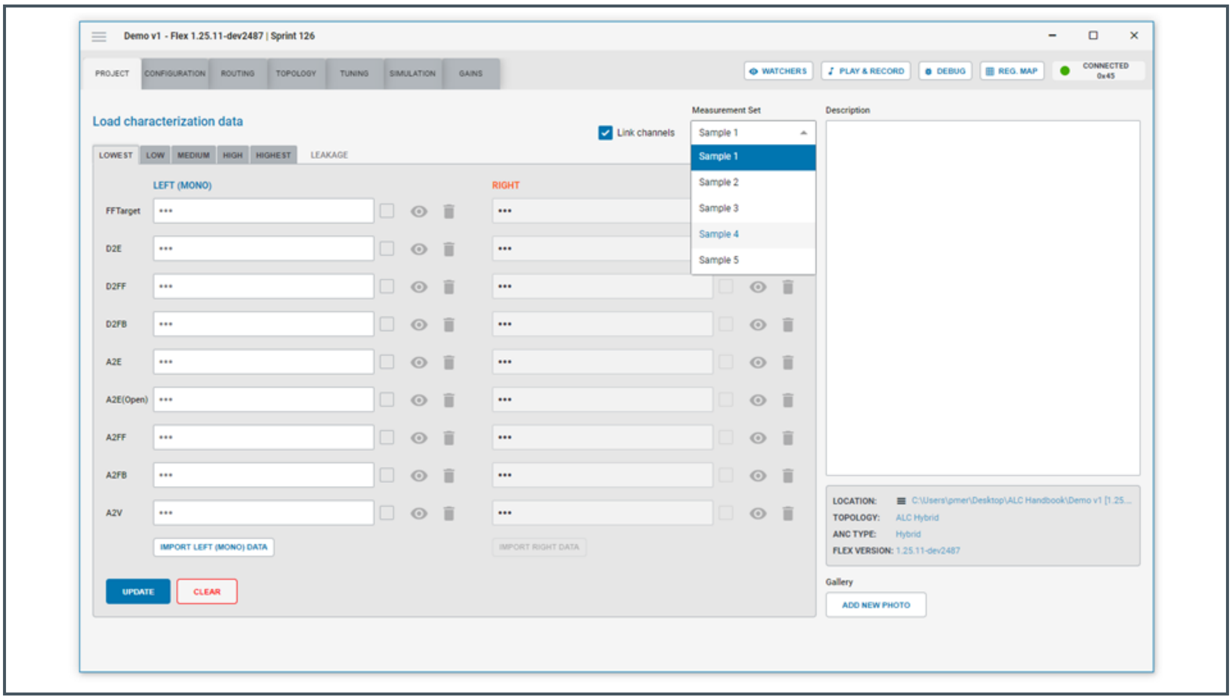


Figure 92:
Measurement Data Import



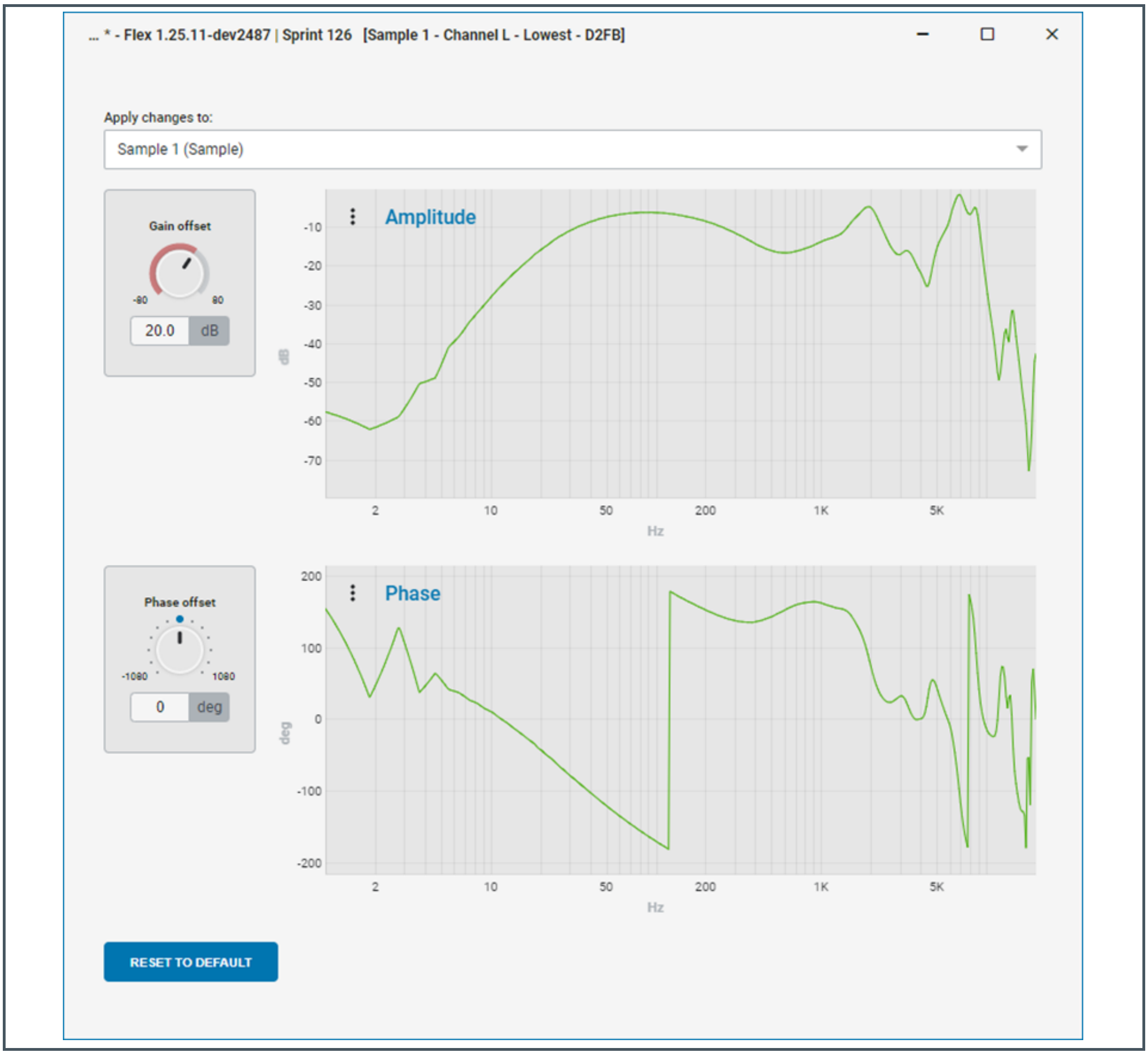
7.4.2 Calibration

Typically, the measurements require calibration. For calibration, click on the eye symbol next to the measurement.

Gain Calibration

The measurement equipment will add a gain to the driver response, which must be compensated. E.g. in the **ams** lab setup, we use an AudioPrecision device set to -20dBV output level and an 0 dB gain headphone amplifier, therefore the gain offset amounts to 20 dB. Since you will probably perform all measurements with the same setup, you can use the "Apply changes to" menu to calibrate all leakages at the same time. If during the measurement, the speaker was wired inverted compared to the application, you can also add a 180° phase offset here. Do not use this calibration for any other purpose; the auto adaption function depends on these measurements being gain accurate.

Figure 93:
Bode Plot from Imported FRs



Phase Calibration

When using microphones, which have different polarity, e.g. using a Knowles SPG08P4HM4H-1 top port as feed forward microphone and a SPH0655LM4H-1 bottom port as the feedback microphone, it will be necessary to invert the microphone signal before it enters the algorithm. You can do this in the routing tab. Because of this, you will need to also invert the corresponding characterization measurements. E.g., when you invert the feedback microphone in the routing tab, you also need to add a 180° phase offset to all A2FB and D2FB curves.

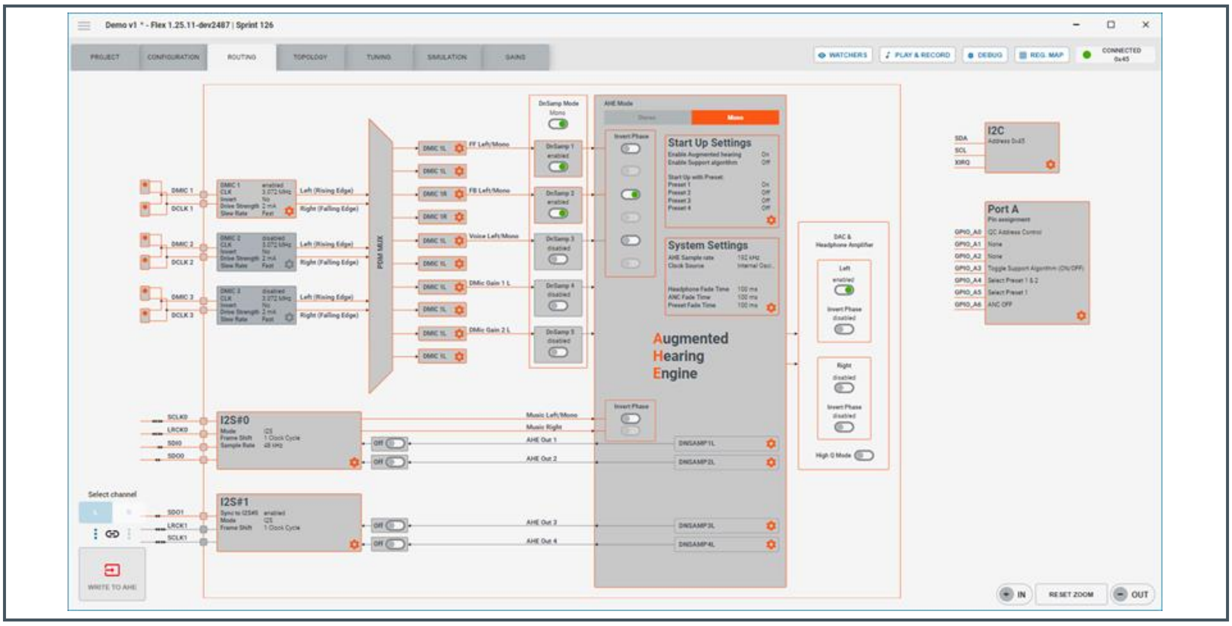
There is no need to calibrate the ear microphone or ambient speaker, since their gains cancel out in all calculations.

7.5 Signal Routing and System Configuration

Before starting the filter design, some settings must be made in the routing tab.

Make sure to select the correct sample rate and microphone clock rate for your project as these parameters affect the filter design. The first step in the filter design will be to design a low leakage feed forward filter and test it in static ANC mode. To do so, turn off the "Support Algorithm" in the startup settings and select only preset 1 on startup.

Figure 94:
Example Routing Settings for Starting an ALC Project



Note: For testing on the EVK, it is convenient to use "Port A". Configure the GPIO pins A3 to A5 for ANC on/off, and preset switching. These pins have pushbuttons connected to them on the evaluation board.

7.6 Feed Forward Tuning – Low Leak Static

Attention

Instabilities can occur during the process of ALC tuning, therefore handle the headphones carefully and mute all filters, which are not required for the current state of tuning. Do not change the filters or configuration while the headphone is on/in the ear.

The lowest leakage state should be quite similar for most people and easily reproducible, which is why the tuning procedure starts as if tuning a static ANC headphone.

Mute all filters in all presets except for the low leakage feed forward filter and tune this filter to match the target. The tuning of course heavily depends on the headphone's acoustic properties. A typical configuration would be:

- Two or more low shelf filters for bass boost
- A peak filter for the resonance around 500 Hz
- A notch filter for the resonance around 2 kHz
- Several additional notch filters to suppress overshoot

Figure 95:
Example of a Lowest Leakage Feed Forward Tuning

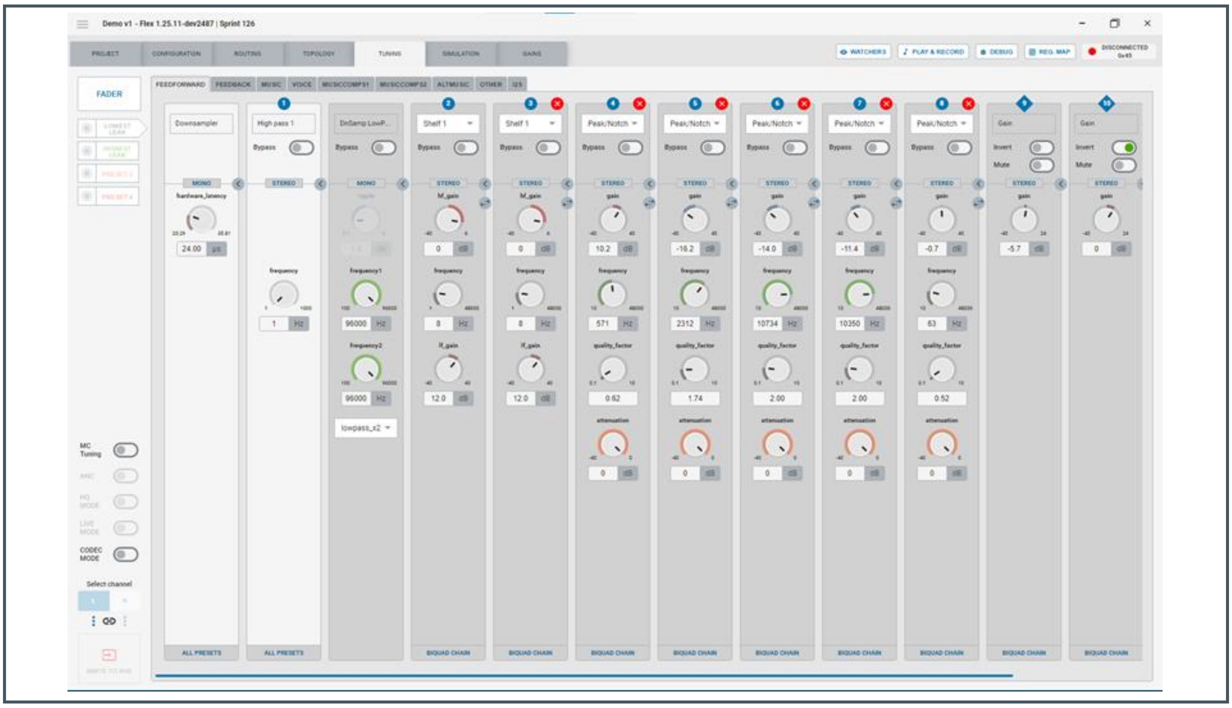
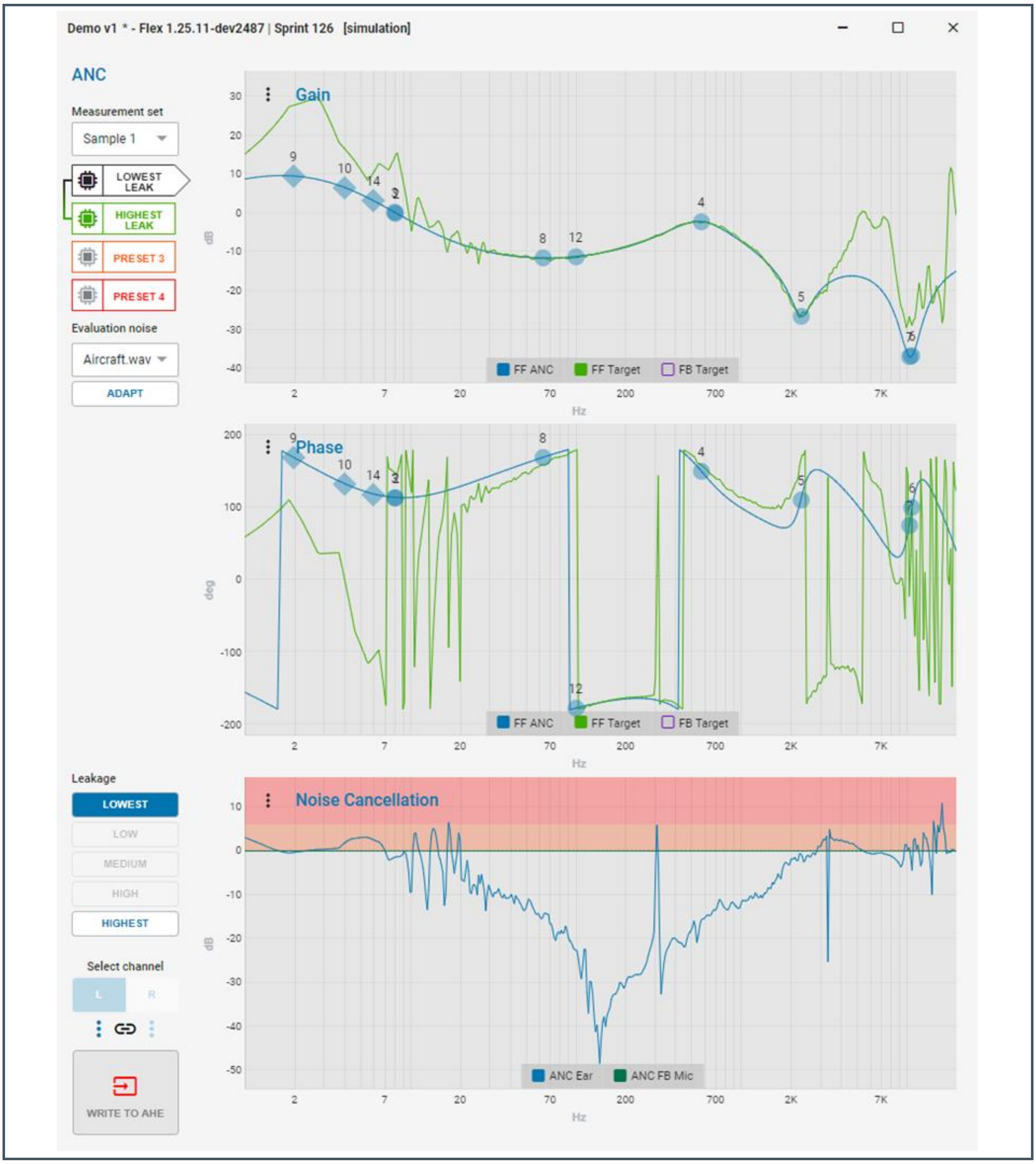


Figure 96:
Target, Filter Response and Performance Simulation For the Filter in the Previous Figure



Once the filter design is complete, you can write the program to AHE and evaluate the ANC performance. Before putting the earphone into your ear, please check that it is stable and that e.g. when placing the earphone in a cupped hand it does not whistle. Then place the earphone in your ear so that it is at lowest leakage. That means you should try to reproduce the lowest leak condition that was used in the measurements. We recommend playing back pink noise for subjective evaluation.

Within this first test, you can already rule out many common mistakes. For example if you do not hear any cancellation, but instead nothing or even a noise boost. This could be due to either a gain offset or an inverted signal. Use the Live Mode to play around with the feed forward gains and inverting the signal. If decreasing the gains improves cancellation, it means that your measurements are not calibrated correctly. If cancellation occurs after inverting the feed forward signal, it means that your measurements were measured with an inverted speaker polarity or the measurement calibration did not account for the microphone polarity. A howling noise could mean that the signal routing is not correct and the feedback microphone signal is going to the feed forward filter.

Information

- Do not invert the signal while the earphone is in your ear.
- Do not use the "Gains" tab for playing around with gains. The gains in the "Gains" tab are meant for production calibration only. These values persist on the chip independent of the configuration. This can lead to mysterious gain offsets and confusion.

At this point, you can also evaluate the quality of the ANC filter. Test if the low frequency and high frequency overshoot is within acceptable levels and if different people experience the same ANC performance. With regular listening setups, it can be hard to spot a low frequency overshoot. We recommend using a very large subwoofer to reproduce low frequency sound pressure levels as they can occur inside a bus or aircraft. If you do not have a very large subwoofer or an aircraft, you might need to purchase a bus ticket.

Information

- If you want to measure the ANC performance, make sure to use a broadband noise test signal, since special signals, e.g. a sine sweep, would distort the adaption.
- The commonly available ear simulators (e.g. HATS) which are, as mentioned before, not suitable for characterizing leaky headphones, are also not suitable for measuring ANC performance in leaky headphones. Please use the **ams** leakage adapter or measure the performance in a real ear by probing the feedback microphone.

7.7 Feed Forward Tuning – High Leakage and Adaption

After proving that the low leakage feed forward filter works, the next step is to design the high leakage feed forward filter. The filter design can be approached similar to the low leakage filter, but you should make sure to not use too much bass boosting. When using too much bass boosting, it could happen that there is not enough signal headroom left to handle situations with excessive low frequency noise, such as sitting on a bus or in a car. Too much means more than 40dB filter gain for typical microphone combinations.

On clicking "Adapt" in the simulation tab, FleX will simulate the noise cancellation performance for the in-between leakages. Use this to check if your two filters also work well when used in parallel.

7.7.1 Testing

Since the low leakage static filter test should have already confirmed that your measurements are gain and phase accurate, it is not necessary to test the high leakage filter in static mode. In addition, it is difficult to perfectly reproduce the highest leakage condition.

Therefore, the next step is to test the basic feed forward adaption. To do so, after tuning the high leakage feed forward filter, enable the support algorithm in the routing tab, enable Preset 1 and 2 in the routing tab and "Disable FB" in the ALC configuration. Then write the new configuration to AHE. Now you should be able to hear a noise cancellation performance at all leakages with the algorithm adapting to the new leakage condition every time the earphone's fit in the ear is changed.

If the algorithm, instead of adapting the gains for maximum noise cancellation, adapts the gains to zero or the maximum, check if the settings in the routing tab are correct. In this case, it is likely that an inverted driver or inverted microphone polarity was not accounted for.

7.7.2 Watchers

Evaluating needs not to rely on subjective listening only, you can use the "Watchers" to observe many internal states of the algorithm in real-time.

For the basic test, the watchers of interest are: the feed forward gains: ff_hl_gain and ff_ll_gain and the good mode state gm_state.

The watcher gm_state ("Good Mode") monitors a state machine internal to the algorithm. There are three states:

Figure 97:
State Transition

State Id	State	Description
2	"Bad"	The algorithm detected that the noise cancellation performance is not good, that means that either the earphone is not in the ear or the algorithm has not yet adapted to the true leakage. Therefore, the feedback filter is turned off to prevent instabilities and the feed forward gain adaption is set too fast mode.
1	"Medium"	This in-between state was introduced to add hysteresis to the state transition, so that in an in-between situation, the user will not hear a frequent on/off of the feedback filter. If the algorithm reaches the state "Medium" and the feedback filter is on, it will stay on. If it was off, it will stay off.

State Id	State	Description
0	"Good"	The algorithm detected that the noise cancellation performance is good, that means that the current leakage estimate is accurate. In this state, the algorithm will ramp up the feedback filter gains and slow down the feed forward adaption in order to converge more precisely.

A state transition to "Bad" can also be triggered by a corner case, such as the user quickly pushing the earphone into the ear canal.

7.7.3 Common Adaption Problems

The following trouble-shooting list shall help in testing the feed forward only adaption:

Slow Adaption

- Often this is because the filters do not match the target well enough or the high and low leakage filters are too similar and do not reflect the acoustic differences between high and low leakage. If this is the case, the algorithm cannot converge towards a leakage value, since both filters fit similarly well to all leakages.
- You should only increase the algorithm adaption speed control if improving the filter design did not improve the adaptions speed. Turning up the algorithm adaption speed control too much will lead to a nervousness of the noise cancellation performance, in the extreme case, the user might hear modulated noise.

Pressure Feeling

- Some users report a pressure feeling if the feed forward filter produces too much low frequency (infrasound) overshoot. Not only is this unpleasant, it is also a hint that the filter will cause problems in environments with high sound pressure levels in the low frequency area.

High Frequency Overshoot

- Above 1 kHz, the target phase deviates depending on the direction of the noise source and the matching between the signal at the eardrum and at the feedback microphone is less accurate. Therefore, the high frequency overshoot might be higher than indicated in the simulation. To solve this, you have to reduce gain above 1 kHz, which usually will be a compromise on the ANC performance in the area below 1 kHz.
- The Live Mode of FleX can be helpful for fine-tuning and directly targeting this overshoot. Make sure to test the high frequency overshoot on several people, because in this frequency range, the individual ear differences are important.

Does Not Reach gm_state_left = 0

- If the algorithm does not reliably reach the gm_state_left = 0 state in all leakage conditions, either the ANC performance achieved with the current filters is not good enough or a corner case prevents the state transition. Make sure to turn off all corner cases in the ALC configuration at this state of the tuning process.

7.7.4 Stability Problems

It can occur that, when going rapidly from highest to lowest leakage, i.e. pushing the earbud quickly into the ear ("push-in"), an instability occurs, because the high leakage filter gain is high enough to cause a feedback loop when used at low leakage. A sub-algorithm named "FF stabilizer" is in place to avoid it. Enable it in the ALC configuration if necessary. If instabilities still occur, you can tune its sensitivity in the "FF stabilizer" section of the ALC configuration. Use the pre-defined watchers (eye-symbol at "FF stabilizer", then click on the eye symbol of FleX's top bar to open the watcher window) to observe the instability metric during a push-in and adjust the thresholds accordingly. If the headphone is very prone for this kind of instability it might be a hint for assembly problems that cause a large acoustic bleed from driver to feed forward microphone.

7.8 Feedback Filter

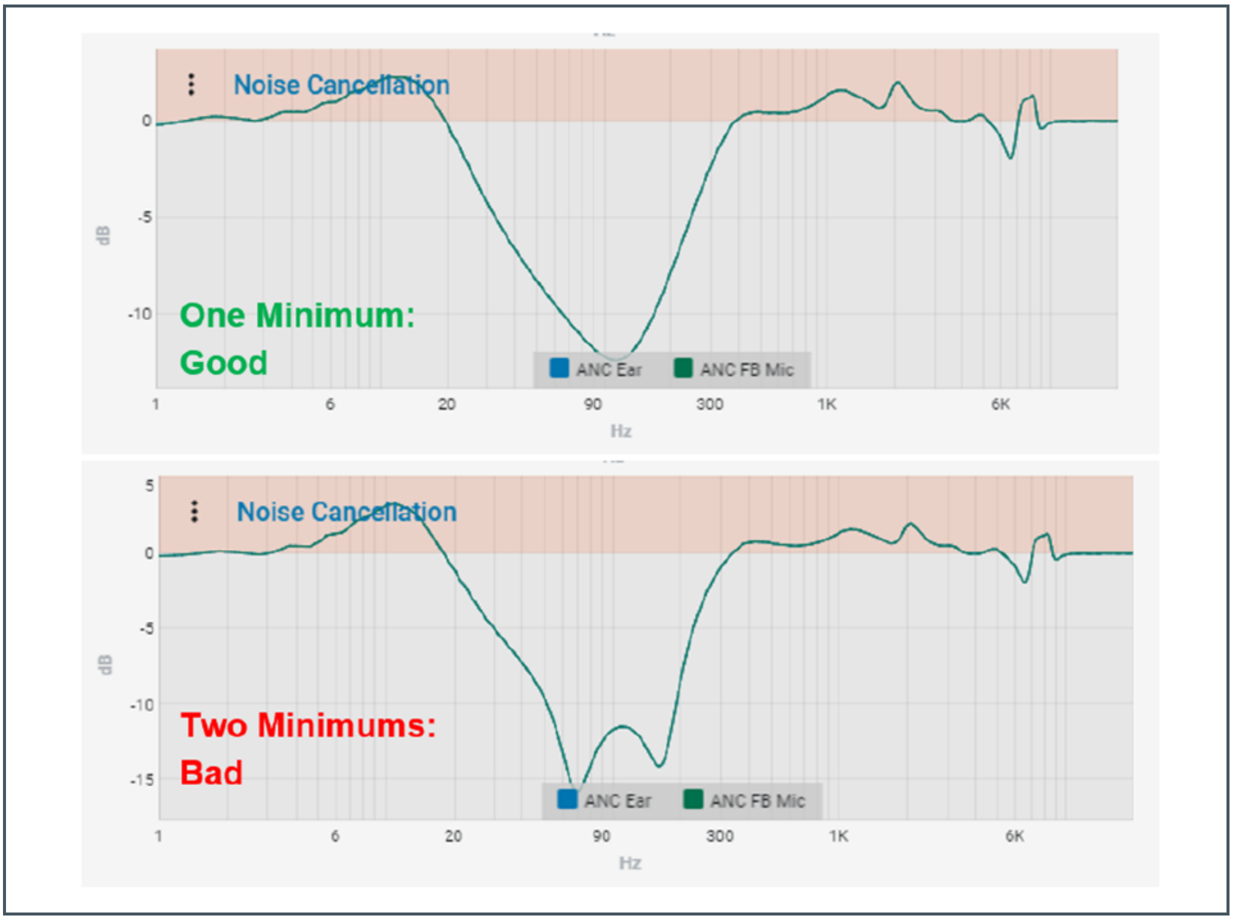
Once the feed forward only adaption tuning is complete and proven to work as desired, the next step is to add feedback noise cancellation.

7.8.1 Feedback Filter Design

In general, the feedback filter can be designed in the same manner like on a non-adaptive headphone, but must meet the following requirements:

- stable above 3 kHz in **all** leakage conditions
- a maximum performance of 10 to 15 dB
- a simple shape of the performance curve with only one minimum
- the downsampler settings must be the same as for the feed forward filter

Figure 98:
Feedback Filter Examples



Always run the auto adaption ("Adapt" button in the simulation tab) before testing a new or modified set of low and high leakage feedback filters. The auto adaption routine calculates several important parameters from the measurement data and filters. Feedback filter gains for the intermediate leakages might be off if the auto adaption routine did not update these parameters.

There is a sub-algorithm, which manages the influence of the feedback noise cancellation on the feed forward noise cancellation. This will only work properly once the music compensation has been tuned, therefore it should be turned off for the initial tests of hybrid ANC. Disable "D2FBM Compensation" and "FB Compensation" in the ALC configuration.

7.8.2 Feedback Stability

Feedback ANC should be stable in all use-cases, which means the earphone should not produce any howling sounds, at least not when it is near or in the user's ear.

The high leakage feedback filter usually uses gains with which the earphone would start to howl if put in the ear at a lower leakage. To counter this, first, the high leak filter should be designed with an increased stability margin but there is also a sub-algorithm in place for detecting instabilities. The proper functioning of this algorithm should be tested for each earphone design. Before testing, activate the corner cases „Enable FB stabilizer", „Enable FB Mic Blocked" and if available on the earphone „Use external proximity sensor".

Test for stability by conducting the following experiments:

- Wear the earphone in the highest leakage condition in which gm_state_left still switches to 0.
- Wear the earphone in various leakage conditions and change the fit.
- Leave the earphone outside of the ear or at a very high leakage, waiting until the feedback gains reached the maximum (see watchers) and then rapidly change the fit to a low leakage. ("Push-ins")
- Perform these test also while speaking
- Repeat these tests with music after the music compensation tuning was completed

No howling or instability should occur in any of these use-cases. If instabilities occur they might appear on isolated frequencies that already exhibit some degree of overshoot in the simulation and the issue can be solved by adjusting the filter tuning, e.g. by placing an additional notch filter. If this does not solve the issue, reduce the overall feedback gain.

FB Stabilizer Parameters

If the instability on push-ins cannot be fixed by filter tuning, you may use the "FB Stabilizer" Section of the ALC configuration and the corresponding watchers (FB Stabilizer Watcher Preset + fb_hl_gain + fb_ll_gain) There you can observe the instability metric during a push-in and then set the attenuation threshold so that the stabilizer reacts more quickly. In addition, you can adjust the stabilizers sensitivity to avoid false positives due to music playback, but usually this is not necessary if the music compensation filters work well. Therefore, you should first revisit the music compensation filter before changing these parameters. Finally, there is the parameter "VAD sensitivity" which you can adjust if the feedback stabilizer has a false positive by being triggered by the user's voice.

7.9 Music Compensation

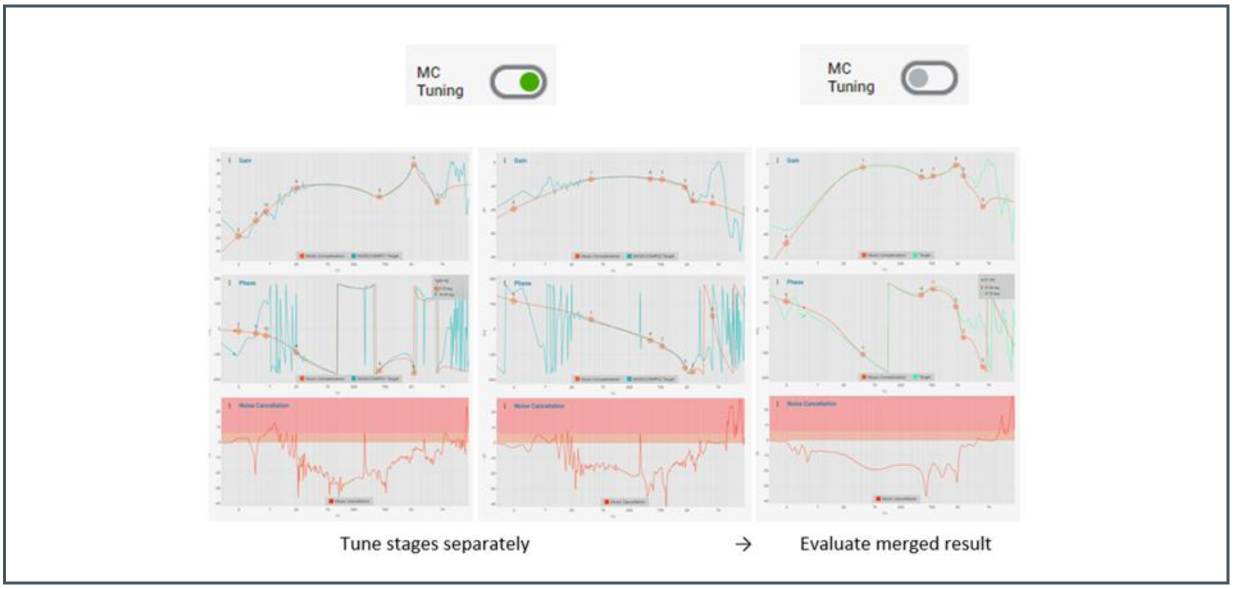
The music compensation filter's purpose is to eliminate the impact that feedback ANC has on the music signal, as well as to eliminate the impact the music signal has on the adaption. Since the feedback ANC filter is adaptive, the music compensation filter must also be adaptive. For acoustical reasons, it is not possible to use the same crossfading scheme here that we use for feed forward and feedback ANC. Instead, there are two filter stages each for highest and lowest leakage and an additional peak filter. The peak filter is fully adaptive and does not require any tuning other than for simulation purposes, but the other four filters must be tuned.

For each leakage, there is a first stage filter, which is tuned to follow the driver response at this leakage and a second stage, which is tuned to follow the passive attenuation at this leakage. The four filters must be tuned separately using the "MC Tuning" mode. Once both stages are tuned, their joint effect can be evaluated by disabling the "MC Tuning" mode.

7.9.1 Music Compensation Filter Design

In the tuning tab, switch to lowest leak, select the „MusiccompS1“ sub-tab and enable „MC Tuning“ in the left column of the tuning tab. Now the simulation tab displays the response of the low leak stage 1 music compensation filter and its target curve. Tune the filter to match the target, in the same manner, as one would tune a feed forward filter. Repeat this process for the highest leak filter.

Figure 99:
Music Compensation Filter Tuning



Next, select the „MusiccompS2“ sub-tab and also match the filters with their respective target. At the lowest acoustic leakage, there will likely be a very damped resonance in the target around 200 Hz. This should be matched using the adaptive Peak/Notch stage (denoted „special“). At the highest acoustic leakage, this resonance will likely be at 1 to 2 kHz and it will be more obvious (less damped). Your setting of this adaptive peak filter serves only for the simulation and will be overwritten by the auto adaption. On AS3460 running ALC it will be adapted in real-time.

Once both filter stages are tuned for both leakages, disable „MC Tuning“ and check if the combined filter also matches the combined target and provides a good cancellation performance. Then execute the auto-adaption to calculate the intermediate leakage performance and algorithm parameters. These parameters also include the adaptive peak stage that means its values might be changed by the adaption.

7.9.2 Music Compensation Verification

In static ANC headphones, any sound coloration due to the feedback ANC's impact not being compensated enough can be easily spotted by A/B listening tests or driver response measurements. This is more complicated in ALC. It is possible to turn off ANC instantly and evaluate the change of the driver response by listening, but it is not possible to instantly turn feedback ANC on again, since the algorithm only activates feedback ANC after converging to a reliable leakage estimate. For this reason, FleX includes a dedicated Music Response Checker Tool, which can be found in the play and record window.

Before using the music response checker:

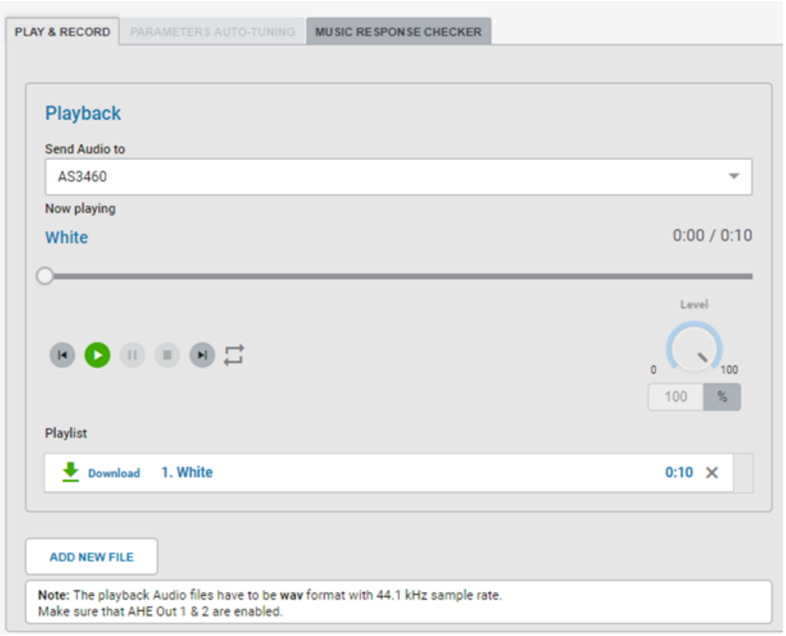
- Make sure the evaluation board is configured so that the Raspberry Pi's I^2S is connected to AS3460.
- Activate I^2S and AHE Out 1 & 2 in the routing tab
- Turn down the music gain by at least 20 dB to avoid speaker and, if you intend to perform the measurement in your ear instead of the **ams** leakage adapter, to avoid hearing damage

The music response checker will require a WAVE file containing white noise named "White.wav" to be uploaded to the Raspberry Pi. Such a file can be found in the FleX program files, with default install parameters the path would be:

C:\Users****\AppData\Local\ams-OSRAM\FleX\resources\resources\evaluation-noise\White.wav

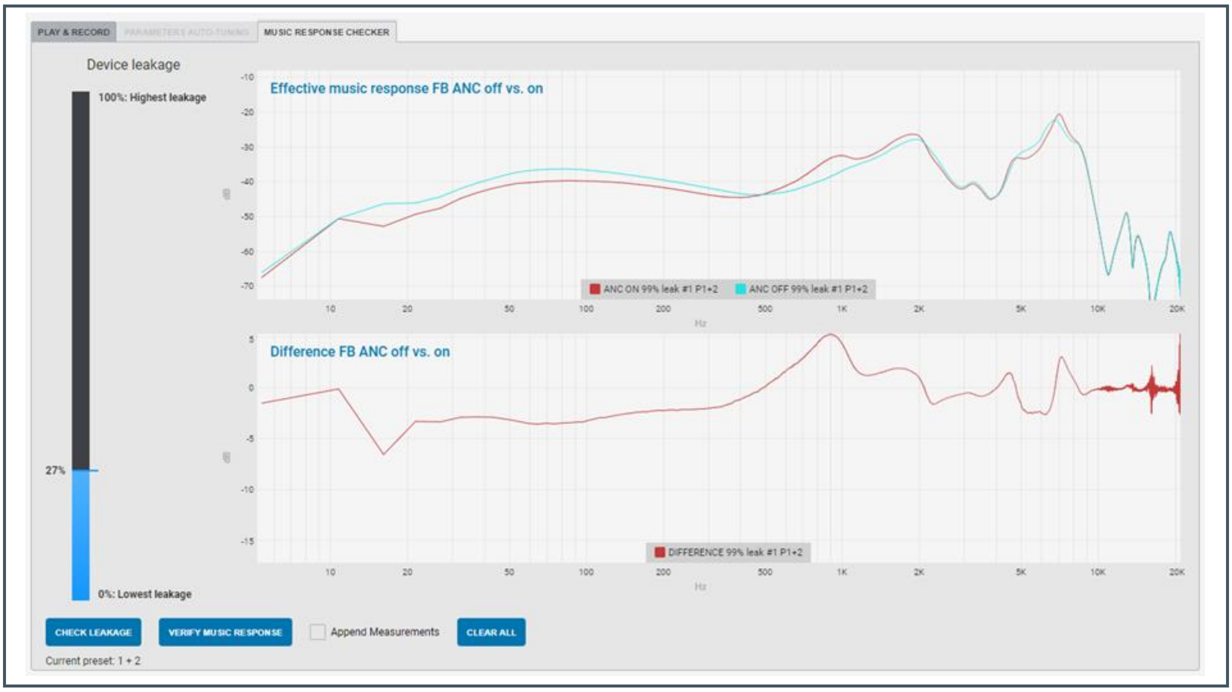
Use "Add new file" in the play and record dialog to upload it:

Figure 100:
Music Compensation Verification Parameters



Place the earphone in your ear or measurement adapter at the leakage that you want to measure and make sure to not change the leakage condition throughout the two following measurement routines.

Figure 101:
Result of the Music Response Checker for a Music Compensation Tuning that Needs Some More Work



Run the "Check Leakage" process to configure the next measurement routine for this particular leakage condition. This process will also check that the noise level is in the correct region and indicate whether further adjustments to the music level are needed. Once the “Check leakage” process has run successfully, click on “Verify music response”. This will run a measurement routine, which checks noise levels and measures the driver response with and without feedback ANC.

The measurement result is presented in two plots. The top plots show the two driver responses and the bottom plot shows the difference between them. Typically, the driver response difference between ANC on/off will not be noticeable if it is less than 2 dB. Repeat this measurement for several leakage conditions. You can overlay several measurements by enabling "Append Measurement". Based on these measurements you can refine the music compensation filters. Make sure to always run the auto adaption before testing a modified music compensation filter.

After the music compensation was tuned, you can activate "D2FBM compensation" and "FB Compensation" in the ALC configuration.

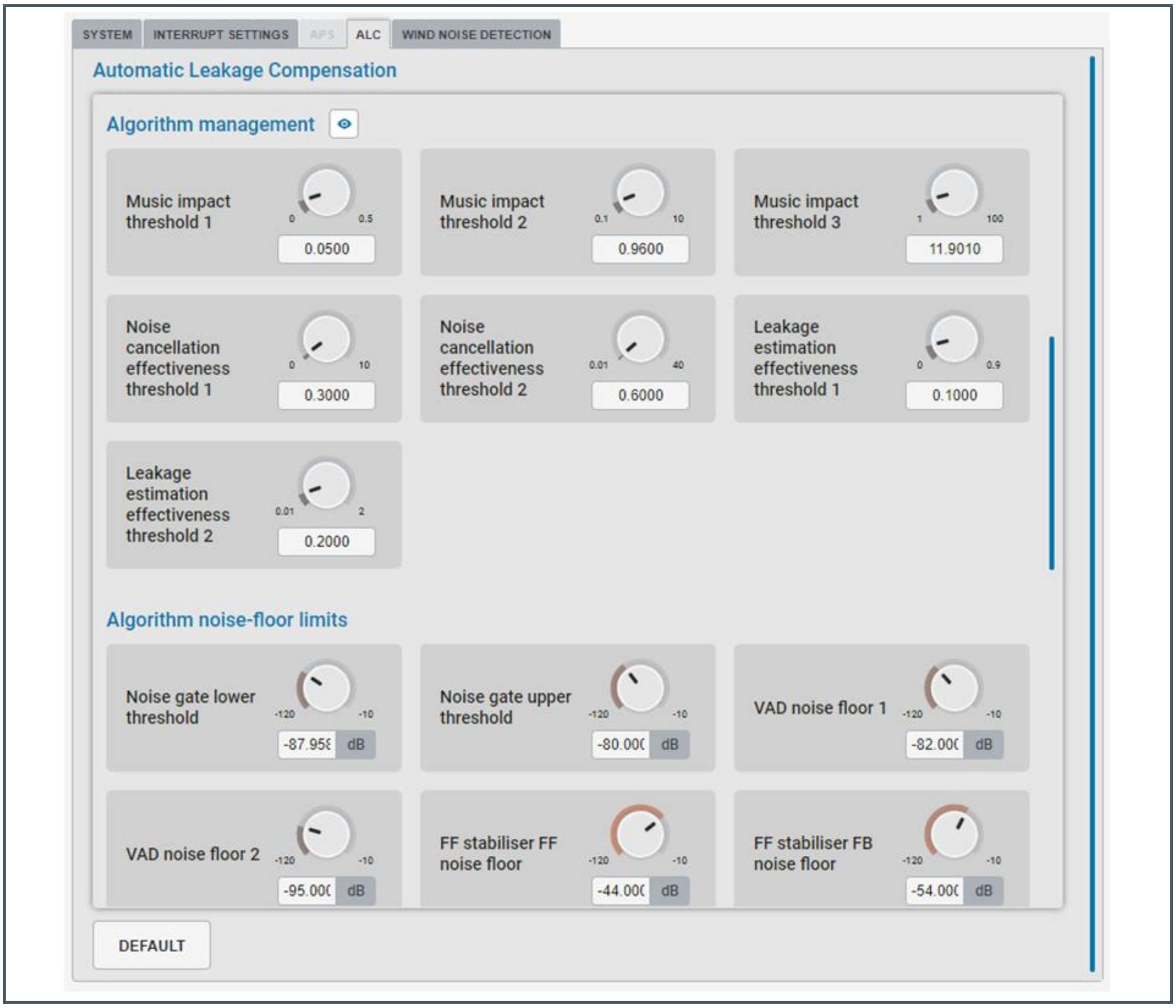
7.10 Algorithm Parameters

A number of parameters can be tuned in the bottom section of the ALC configuration. Their default parameters will work for most headphone designs and you only need to tune them if there are problems with the algorithm behavior and you find that these cannot be fixed by improving the filter tuning.

7.10.1 Algorithm Management

The algorithm calculates an estimate of the noise cancellation effectiveness. If the noise cancellation is effective, the algorithm knows that its current leakage estimate is close to the real leakage and it is therefore safe to activate feedback ANC. If your headphone does not provide a good noise cancellation, you might need to increase these thresholds. Disable feedback ANC, and at different leakages, monitor the noise cancellation attenuation and note at which value it stabilizes. Set the noise cancellation effectiveness threshold 1 75% of this value. Threshold 2 should be around 150% to 200% of threshold 1.

Figure 102:
Algorithm Management Window



The algorithm's estimate of the current leakage condition is based either on the ratio of the adapted feed forward gains or on the observed driver response. Which method is used depends on the music to noise ratio. The thresholds at which the algorithm switches to the different methods are configured using the music impact thresholds. Below the first threshold, it is assumed that there is no music and above the second, it might switch to the different method. Lower the second threshold if you observe that the feed forward gains become unstable on music playback. If you observe that the leakage estimation changes in a stable way when increasing the music level without changing the leakage, you should tune the leakage estimation effectiveness Threshold. The second threshold should be 200% of the first.

7.10.2 Algorithm Noise Floor Limits

Only touch these if you observe problems due to false positives of the respective corner cases. Aside from the pre-defined watcher presets, you can observe the behavior of all sub-algorithms by searching for their watcher. E.g. to observe the behavior of the noise gate, search for noise_gate and find: noise_gate_state_left, noise_gate_lo_thrshld, noise_gate_hi_thrshld and noise_gate_level left. Use these to check if the level exceeds the thresholds even in an environment, which has sufficient noise for the adaption to converge. Use the same workflow to adjust the VAD noise floor (adaption paused by VAD even though the user is not speaking) or the FF stabilizer. Note that it is very unlikely that it will be necessary to change these limits, since they are independent of the headphone's acoustic properties.

7.10.3 AGC

The AGC tightly integrates with the adaption algorithm; therefore, settings such as time constants are governed internally. Adjust the "Near clip threshold" to control the level onset of gain adjustments. Below this threshold, but close to it, the AGC will have a fast recovery and further below the AGC will have some effect. Above the threshold, suppression of the different ANC paths starts. Depending on the adaption state, the AGC might try to suppress the frequency spectrum non-uniformly. The low-pass filter in the "AGC" tab of the highest leakage preset controls the frequency region that this will happen in.

The default settings for the "Outer band energy threshold" and "Adaption guard threshold" should suit most devices. However, the following scenarios might suggest some fine-tuning is required:

- If low frequencies highly dominate the ambient noise and adaption does not stabilise, disable feedback noise-cancellation. If the issue persists, reduce the "Outer band energy threshold" until the system is stable.
- If loud transient ambient sounds are causing unstable adaptive gain adjustments, such as clapping or door slams, then the "Adaption guard threshold" need adjusting. However, please make sure AGC is producing the desired behaviours before adjusting the adaption guard. If adjustment is required, reduce the threshold until the audible artefacts disappear.

7.10.4 Other Parameters

All other parameters of these corner case algorithms as well as the parameters of the corner case algorithms that were not mentioned so far do not require any tuning. The remaining corner case algorithms are:

- Outer Energy

- FB Mic Blocked
- Adaption Guard

They are independent of the earphone's acoustic properties and filter tuning and therefore do not require any tuning. Explanations on what these corner case algorithms do is provided by the tooltips. The tuning instructions for the wind noise detection algorithm are provided in a separate document.

7.11 ELF Stage

In some earbud designs, the driver's low frequency response diminishes strongly with increasing leakage. Especially if a small speaker and no basstube was used. An extra adaptive shelf filter can be activated in ALC to improve performance on these types of earphones.

To use it, enable the "ELF adaption" in the ALC configuration. A new filter stage, denoted "Special", will then appear in the tuning tab at the end of the feed forward filter chain. You can then try to run the auto adapt function. However, if the process times out with the warning: “ELF band pass could not produce required attenuation levels”, then please try to increase the extra low-frequency attenuation or adjust the adaption region, these settings can be found at "Adaption Region" in the ALC configuration.

If the adaption with ELF stage still does not succeed, it usually means that the ELF stage does not provide a benefit for this headphone.

If the “Adapt” process does run successfully with the ELF stage enabled, but the noise cancellation performance at intermittent leakage is not satisfactory, adjust the feed-forward high-pass filter cut-off frequency marked as “Special”. Then, try different frequency settings running the “Adapt” process each time until there is a satisfactory level of noise-cancellation performance at intermediate leakage.

8 Dual Chip Operation

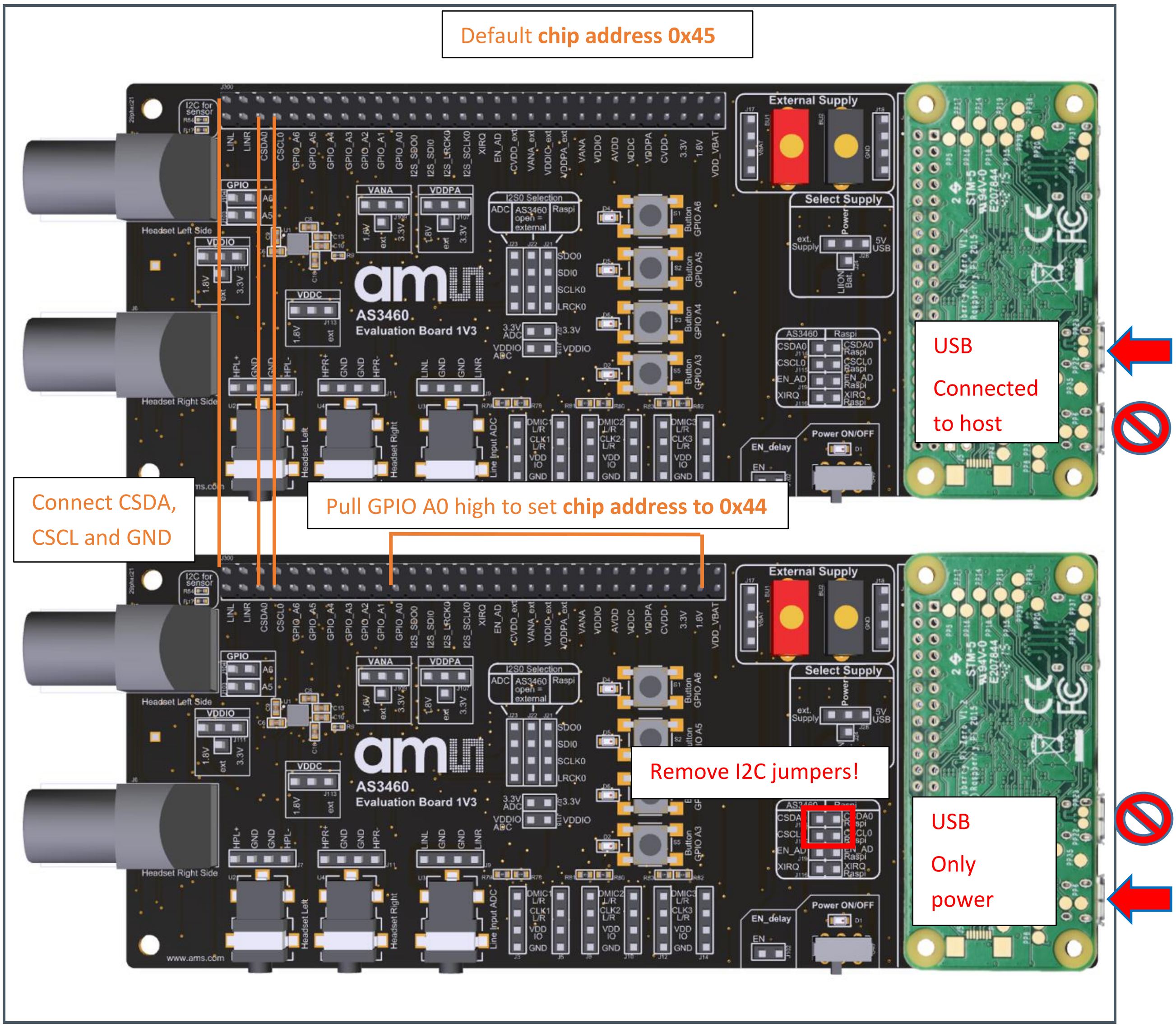
8.1 How to Operate in Dual Mode?

8.1.1 Basic Setup

Connect the 2 boards as shown in Figure 103. The board with the GPIO A0 pulled high will change to address 0x44 after power cycle the chip.

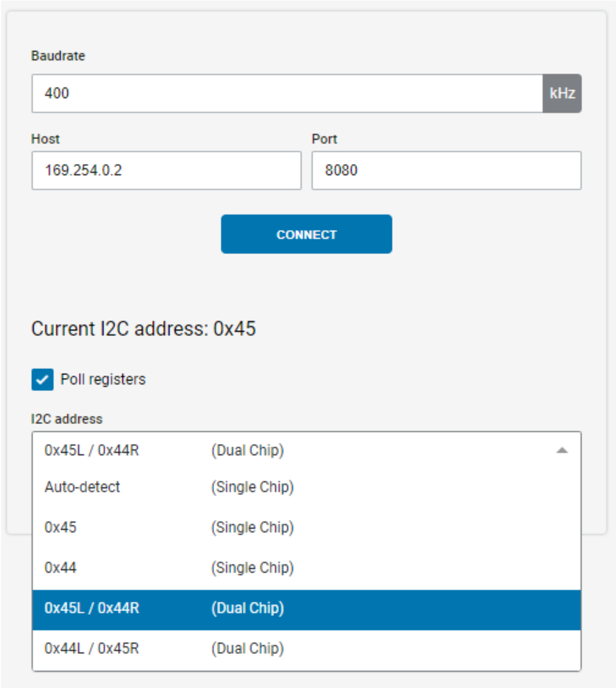
8.1.2 Using FleX with Dual Mode

Figure 103:
Dual Board Configuration



The drop down menu in the connection tab can be used to manually switch between the 2 chips. When using FleX in dual mode one of the Dual Chip items must be selected.

Figure 104:
Dual Chip Address

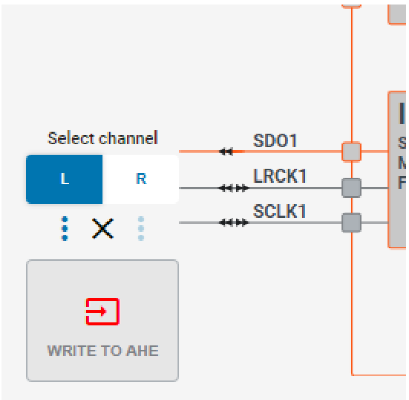


Both chips must be set to AHE “Mono” mode. (When Dual mode is selected it is also shown in the AHE Mode selector)

Figure 105:
Dual Mono Mode in FleX

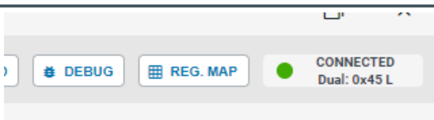
Depending on the selected channel (left or right) the corresponding filter settings will be written to the pre-defined chip (e.g. 0x45 L, 0x44 R)

Figure 106:
Left / Right Selection



Note: Only the filter parameters are synchronized with the L/R selector when writing to AHE. The routing and all other chip settings are written to the currently selected chip. The status of the valid connection can be seen in the connected symbol/tab.

Figure 107:
Connection Status



9 I2S Swap Function

The I^2S swap function allows to flip the left and right I^2S (music) input channel. This feature will be needed if 2 AS3460 in mono mode are connected to e.g. 1 stereo Bluetooth device. Because the AS3460 in mono mode is always using the Left channel it is necessary to route the right Music input signal to the left channel.

Figure 108:
I^2S Channel Swap

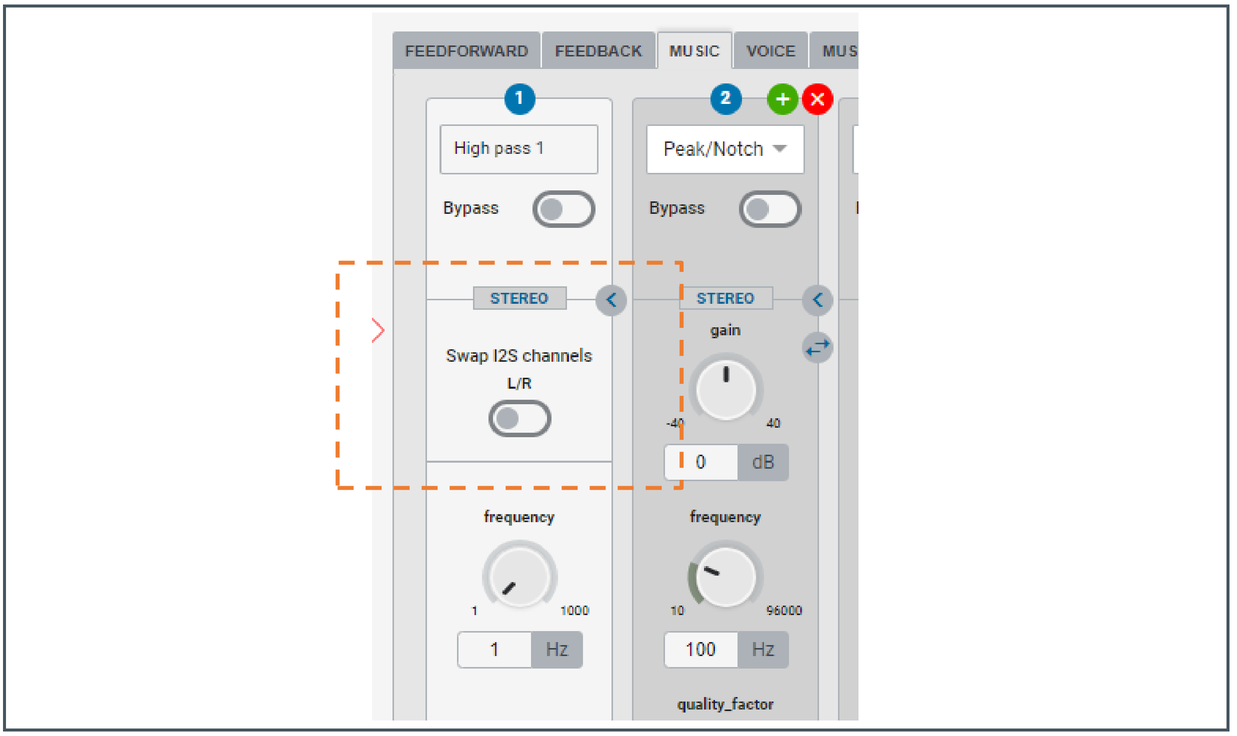
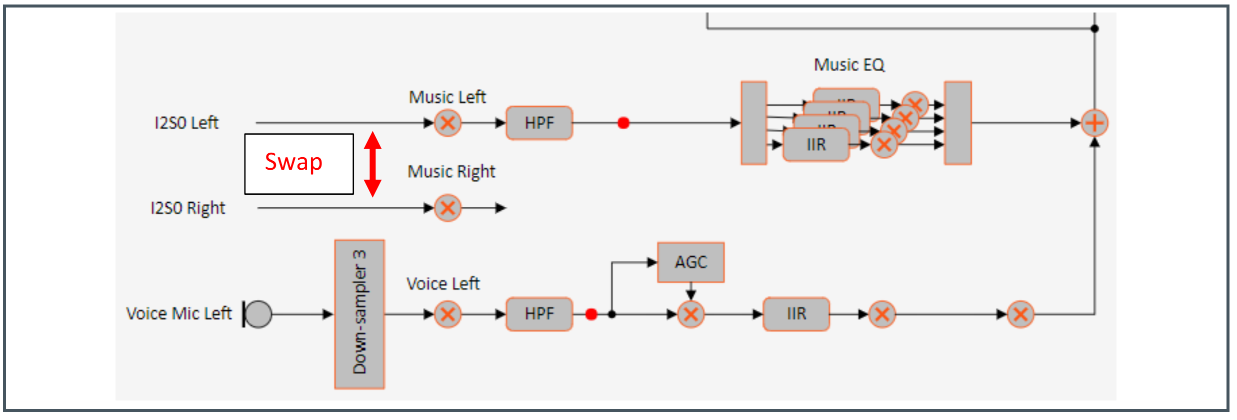


Figure 109:
I^2S Swap Indication



If 2 chips are connected the swap function will be stored to each chip separately. TDM will swap channels 0 and 1.

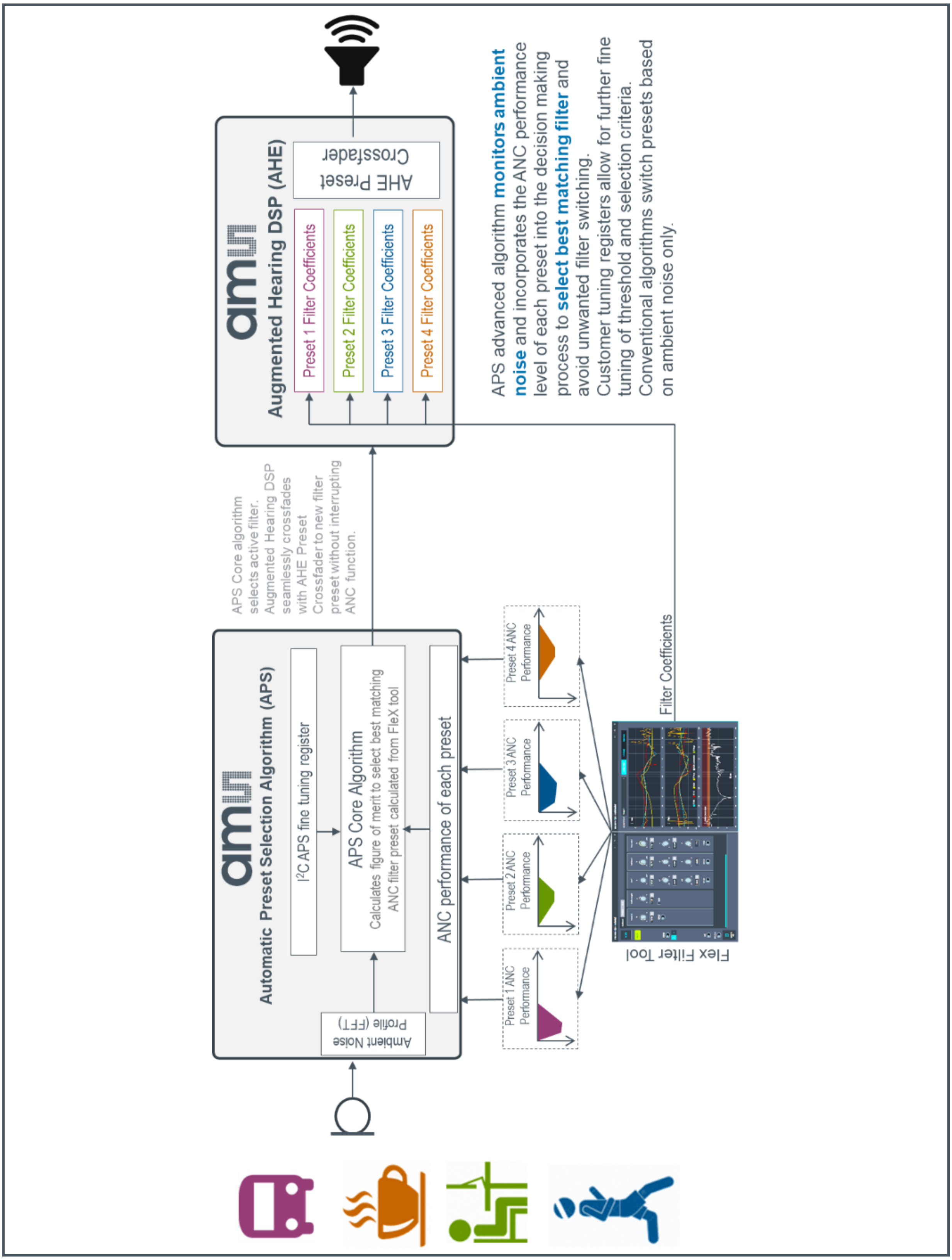
10 Automatic Preset Selection (APS)

10.1 How Does Automatic Preset Selection (APS) Work?

10.1.1 Basic Functionality of the APS Algorithm

For details, please refer to Figure 110 below.

Figure 110:
APS Algorithm



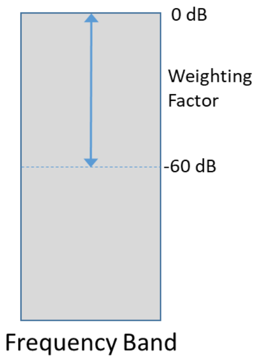
The APS Algorithm selects the best ANC-Filter for a given noisy environment based on an analysis of the Feedforward microphone signal. That means that it analyses the frequency bands of the noise signal and selects the preset with the filter that is mostly cancelling the noise in the frequency band where the most energy of the noise signal is located.

Base for the calculation of in which band the most noise cancellation is done is the simulation.

The algorithm looks for the dominant signal in the 4 bands: 30 Hz, 100 Hz, 300 Hz and 1000 Hz.

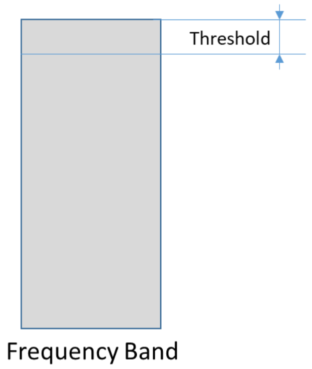
You can choose the most dominant frequency band(s) by scaling the weighting factors, each band can be weighted by a factor of -60 dB to 0 dB.

Figure 111:
Weighting Factor



The Threshold defines when the preset will be switched. That means, that if the calculated energy of the frequency band drops below that threshold, the preset will be switched.

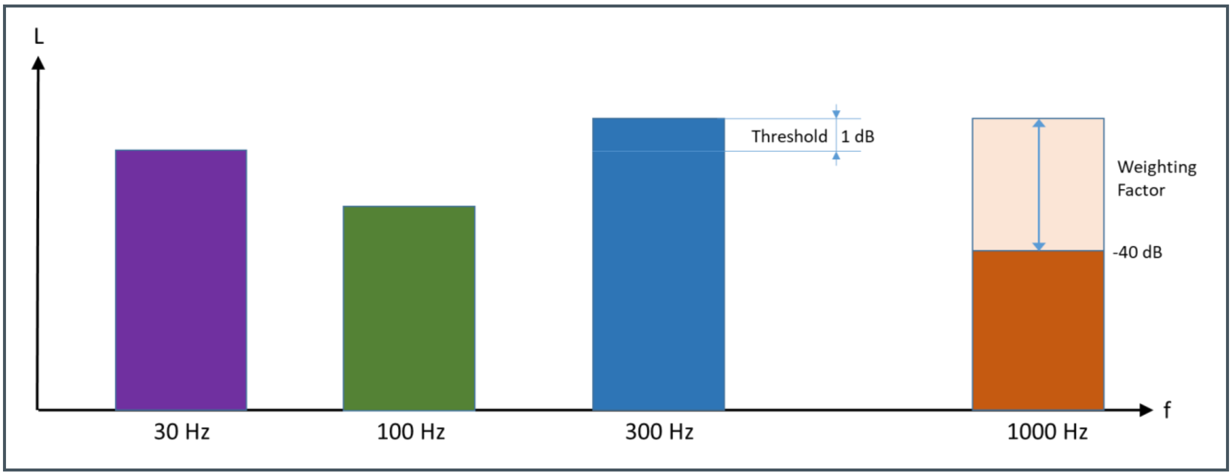
Figure 112:
Threshold



Example:

In the following example the 1000 Hz band would be the most dominant, but since the weighting factor was scaled to -40 dB the 300 Hz band becomes the most dominant. The threshold factor is set to 1 dB, therefore if the level of the 300 Hz band drops by 1 dB the next band will be the most dominant, here this Is the 30 Hz band.

Figure 113:
Threshold and Weighting Factor Example



10.1.2 Selectable Presets

You can select, if the algorithm switches between 2, 3 or 4 presets. The Algorithm automatically detects which frequency band is corresponding best to the filter that was tuned in each individual preset.

10.1.3 APS and Manual Preset Switching

If you are using the APS-Algorithm it is not possible to switch through the different presets manually, since the APS-Algorithm controls the Preset Management.

10.1.4 APS and Monitor Mode

It is not possible, to include a Monitor mode preset into the APS-Algorithm. To use the Monitor mode preset, you have to deactivate the APS-Algorithm and after the Monitor mode was used the APS-Algorithm has to be restarted again.

10.1.5 APS and ALC

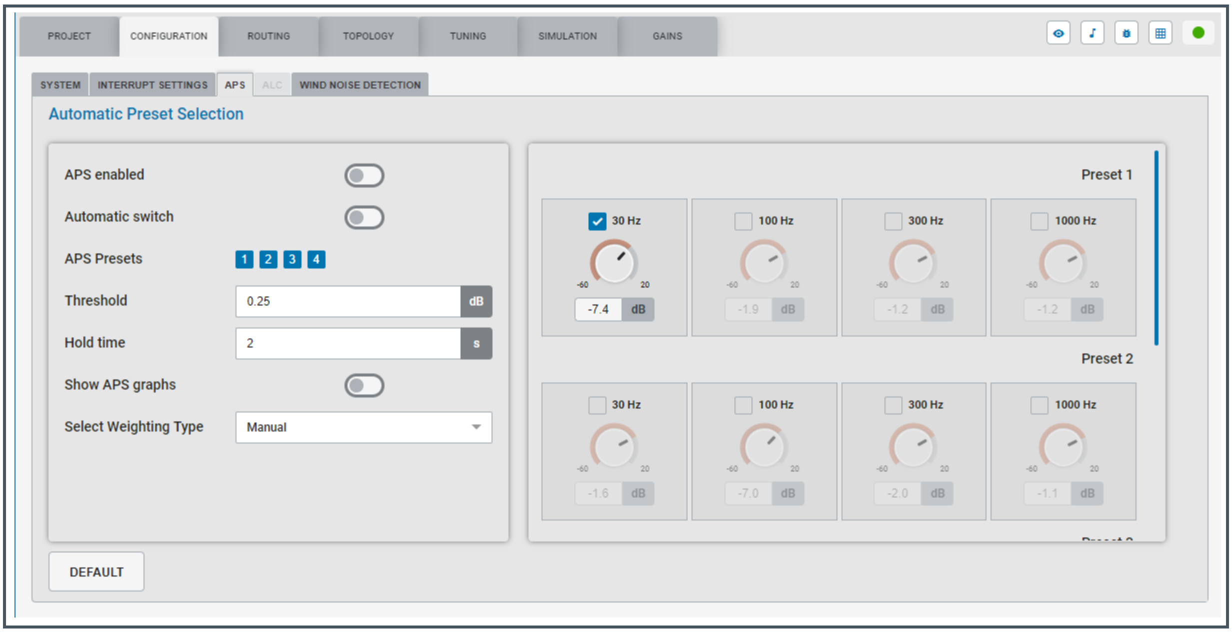
APS is a functionality in the ANC-Hybrid Topology, therefore it is not possible to use ALC at the same time, since this needs the ALC-Hybrid Topology.

10.2 Using APS in FleX

APS can be activated and controlled via the “APS”-Tab in the “Configuration”-Tab.

Note: The “Enable support Algorithm” – Bit must be activated to run APS.

Figure 114:
APS-Tab



The parameters are explained in the following table.

10.2.1 Parameters

The run-time controls for the APS block are summarized in the following table.

Figure 115:
APS Parameters

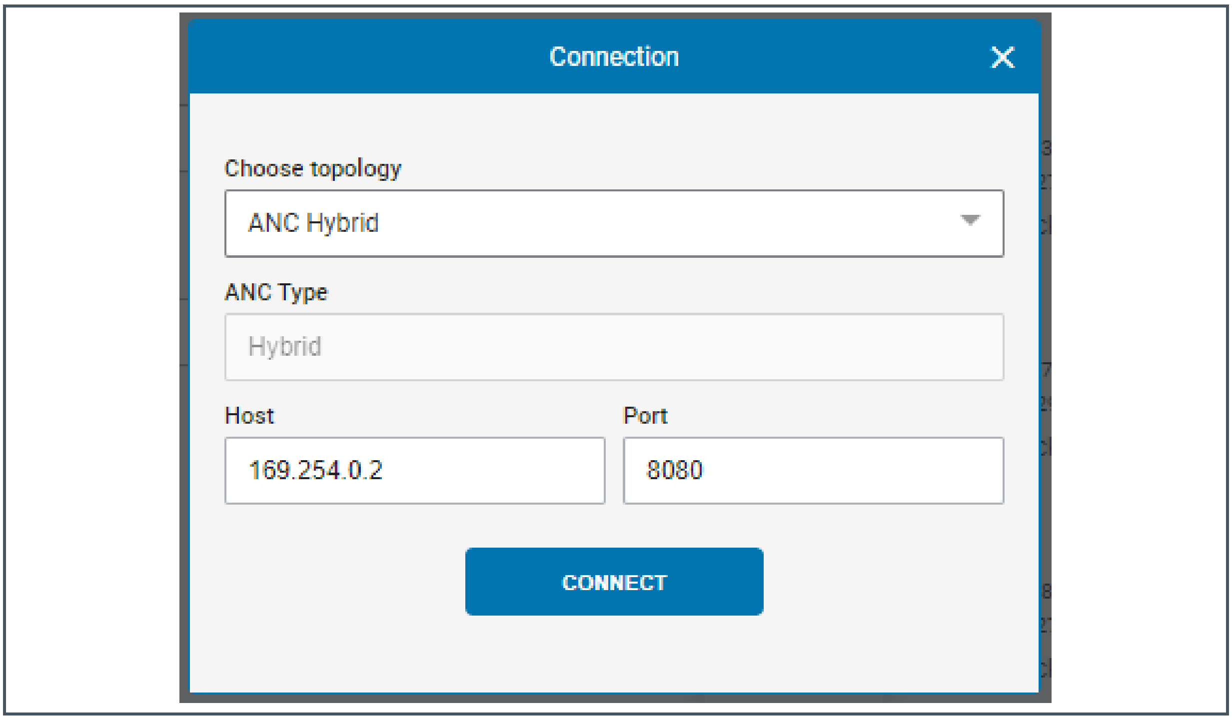
Control Name	Range			Unit	Description
	Min	Typ	Max		
APS Presets	2	3	4	-	The number of presets assigned to APS. Selected Presets will be used for APS. Other presets are presumed to be assigned for other purposes (for example monitor mode)
Threshold	0.1	0.25	6.0	dB	The threshold that must be exceeded before switching presets
HoldTime	0.5	1.0	10.0	seconds	The time, after entering a new preset, when the algorithm can consider changing preset.
Band weighting 1	-60	0.0	0.0	dB	Manual weighting for frequency band 1. Supplements ‘A’ weighting and passive weighting.
Band weighting 2	-60	0.0	0.0	dB	Manual weighting for frequency band 2. Supplements ‘A’ weighting and passive weighting.
Band weighting 3	-60	0.0	0.0	dB	Manual weighting for frequency band 3. Supplements ‘A’ weighting and passive weighting.
Band weighting 4	-60	0.0	0.0	dB	Manual weighting for frequency band 4. Supplements ‘A’ weighting and passive weighting.

10.3 Simple APS Example

10.3.1 Open APS Project

The first step is to open a new project with a Hybrid ANC Topology.

Figure 116:
Opening APS Example Project



10.3.2 Load Frequency Response with All Zeros

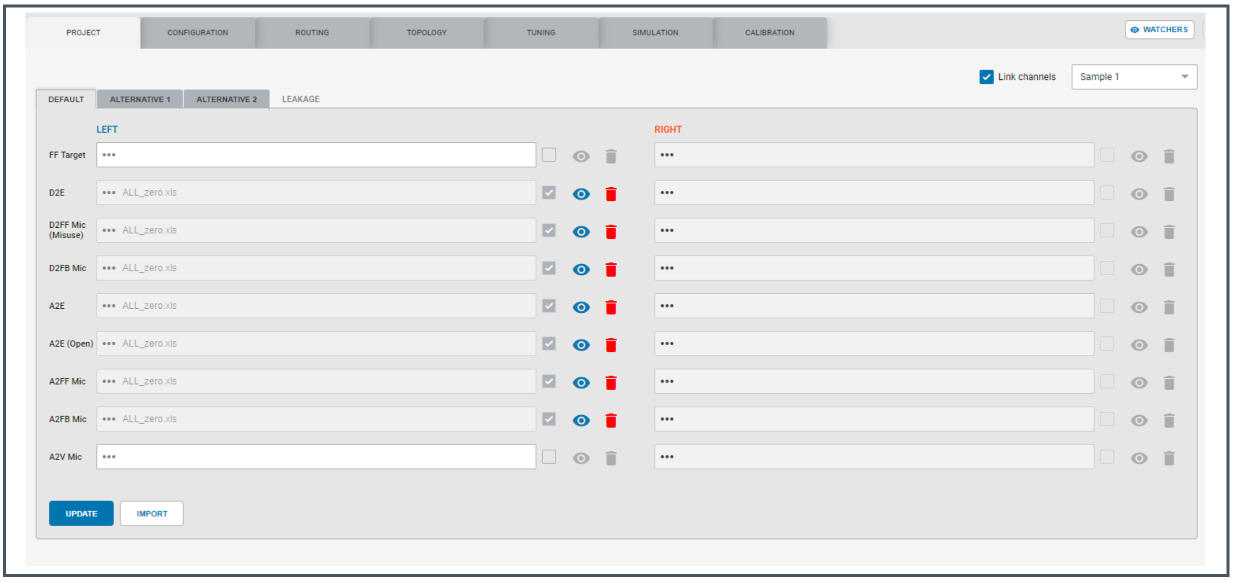
For this step, you need the Excel-File with zeros at all frequency steps and at all phase steps (ALL_zero.xls) you received with this description or you can request at the PS Audio Apps Team.

Please load this file to the following characterization files in the Project Tab in FleX:

- D2E
- D2FF Mic (Misuse)
- D2FB Mic
- A2E
- A2E (open)
- A2FF Mic
- A2FB Mic

In the end, it should look like this:

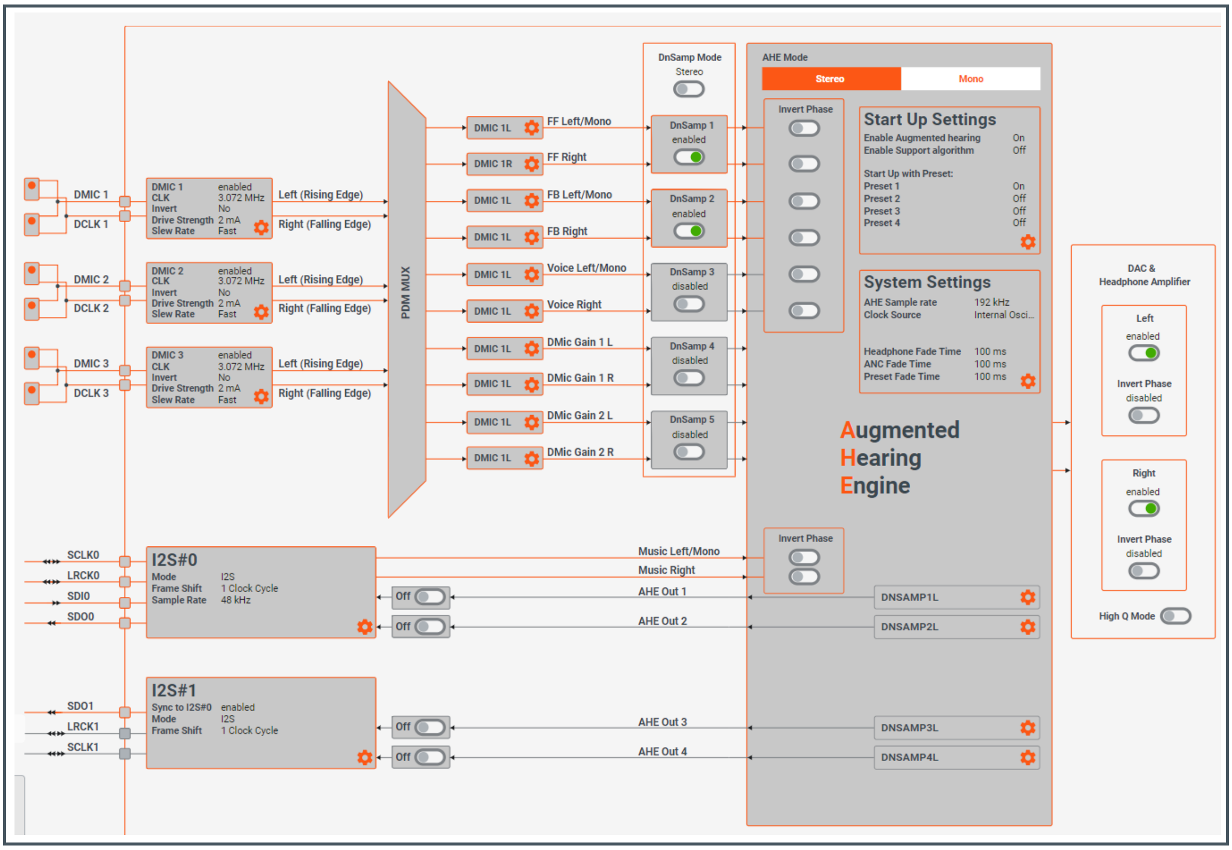
Figure 117:
APS Example Project Tab



10.3.3 Routing

In the routing tab, please route DMIC1L to FF Left and DMIC1R to FF Right as you see it in the following picture.

Figure 118:
APS Example Routing Tab



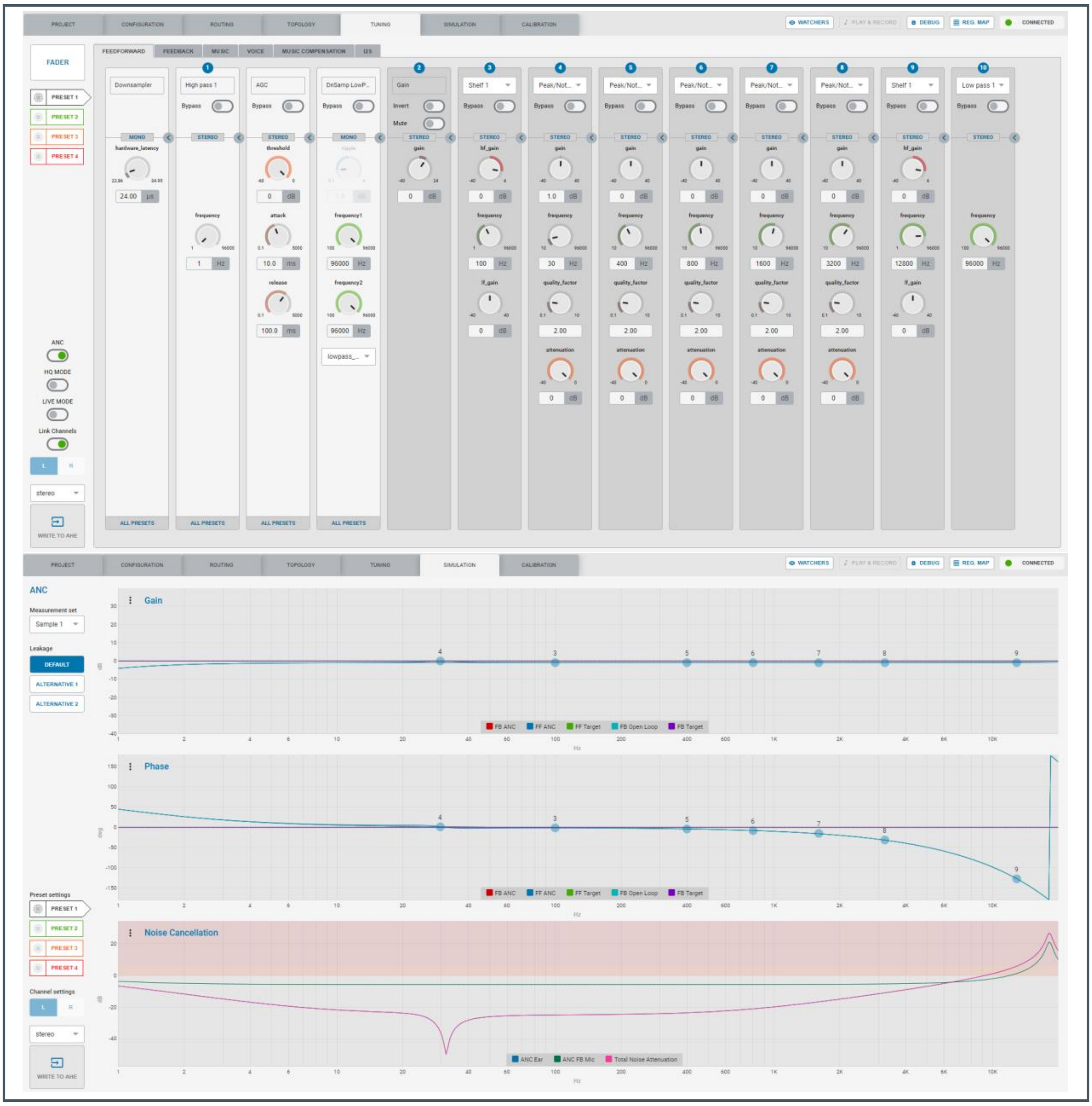
10.3.4 Tuning

In the Tuning Tab we now prepare the 4 presets so that we can use it afterwards for the Automatic Preset Selection (APS).

Preset 1

In Preset 1, apply a Peak Filter with a Q of 2.00 to a frequency of 30 Hz with the gain of 1 dB. This creates a noise cancellation at 30 Hz with about -45 dB.

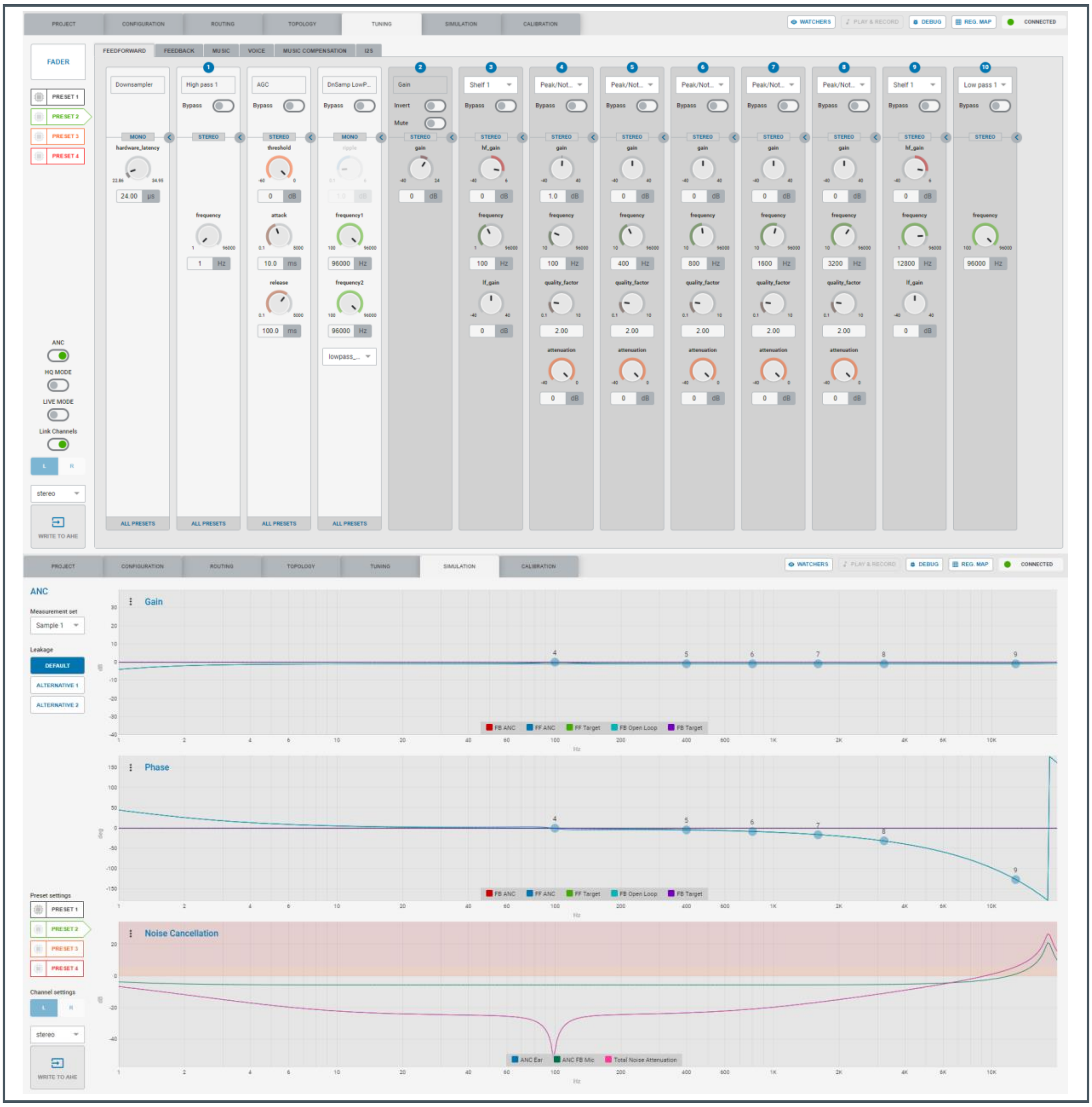
**Figure 119:
APS Example Preset 1**



Preset 2

In Preset 2, apply a Peak Filter with a Q of 2.00 to a frequency of 100 Hz with the gain of 1 dB. This creates a noise cancellation at 100 Hz with about -45 dB.

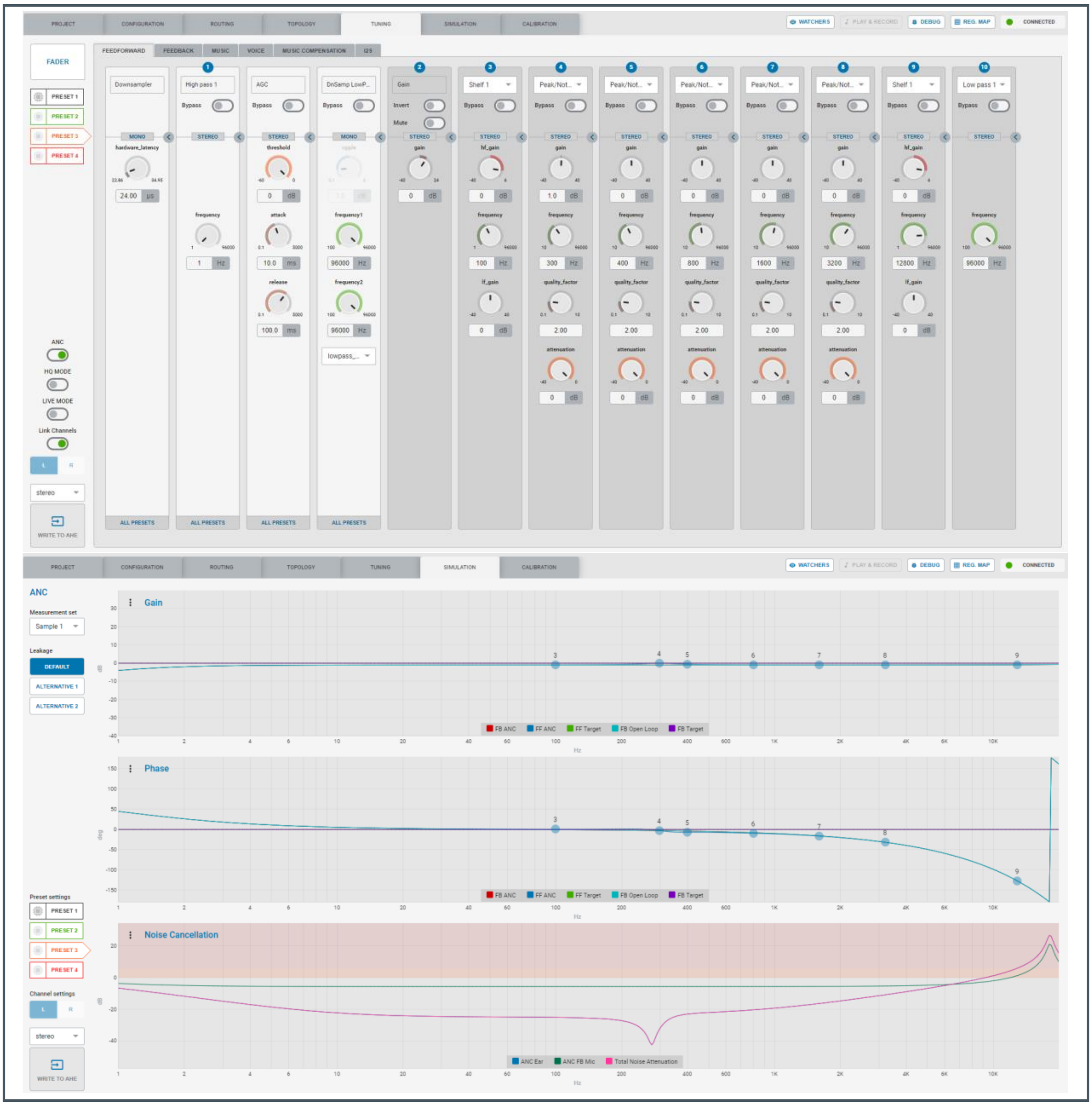
Figure 120:
APS Example Preset 2



Preset 3

In Preset 3, apply a Peak Filter with a Q of 2.00 to a frequency of 300 Hz with the gain of 1 dB. This creates a noise cancellation at 300 Hz with about -45 dB.

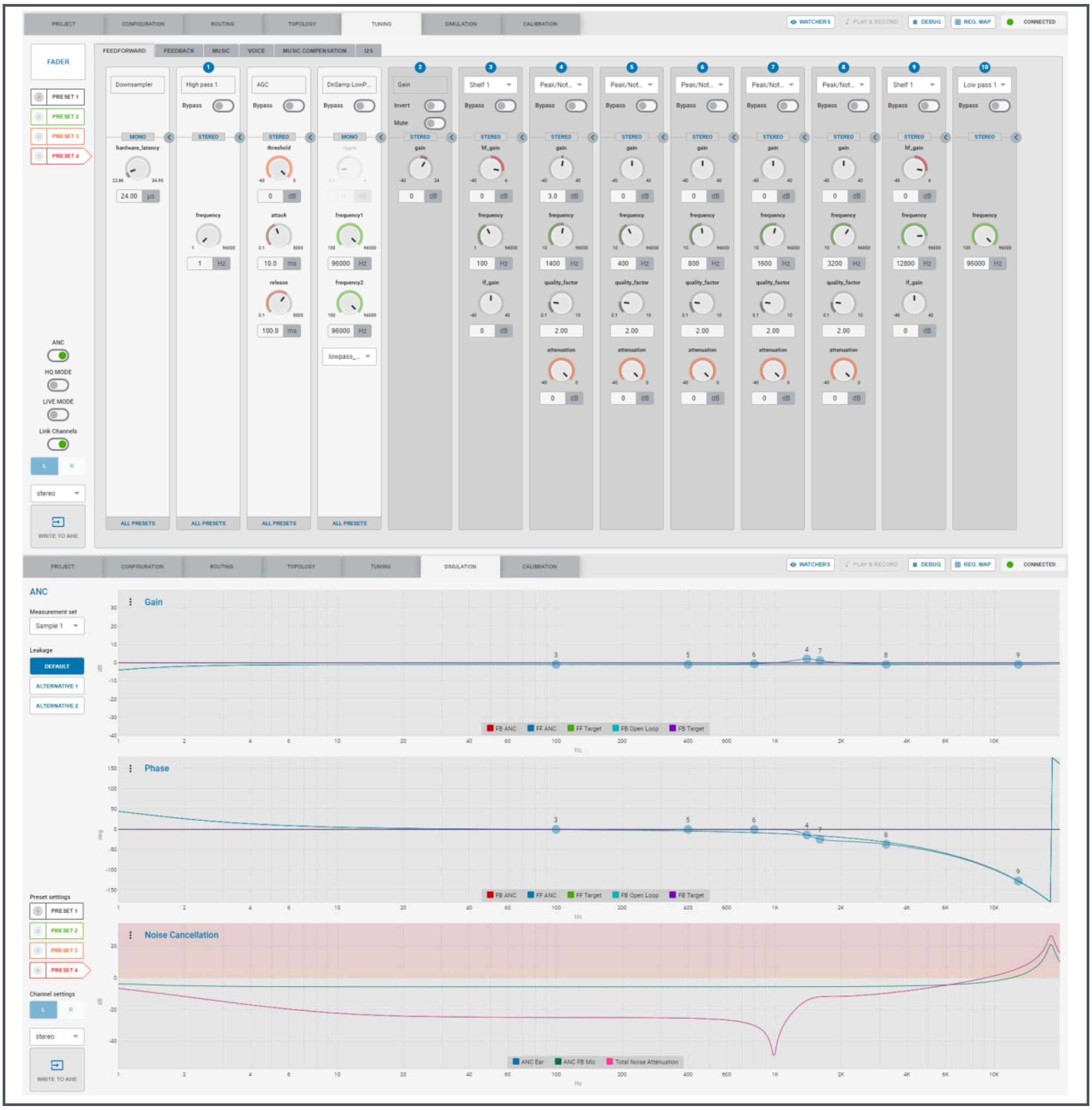
Figure 121:
APS Example Preset 3



Preset 4

In Preset 4, apply a Peak Filter with a Q of 2.00 to a frequency of 1400 Hz with the gain of 3 dB. This creates a noise cancellation at 1000 Hz with about -45 dB.

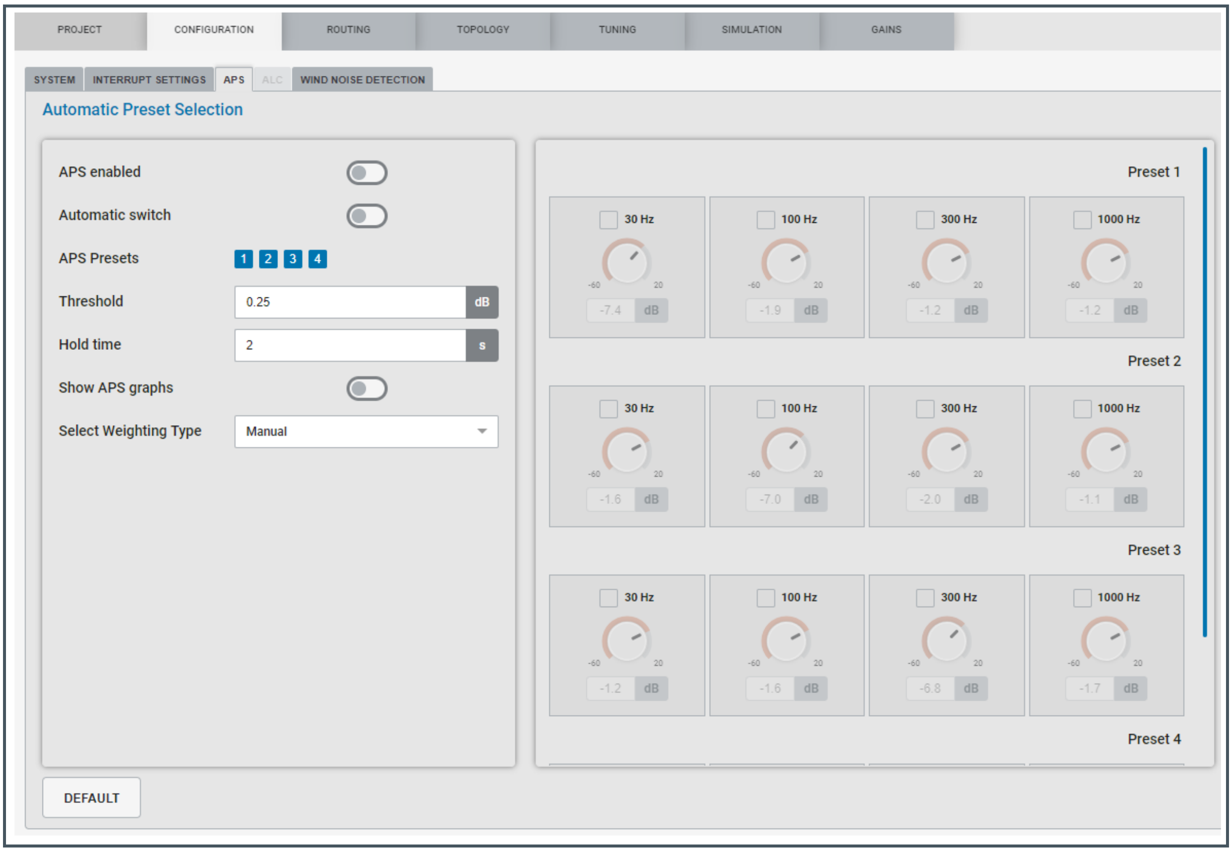
Figure 122:
APS Example Preset 4



10.3.5 Starting the APS Algorithm

The last step is to activate the APS Algorithm with the Standard settings as shown in the following figure. This can be done in the “Configuration” Tab at “APS”. Just activate the switch “APS enabled”. If you are not sure that you have the default values, just press the “Default” button before activating APS.

Figure 123:
APS Example Start Algorithm



10.3.6 Testing the APS Algorithm

If you now feed a sine signal with 30 Hz, 100 Hz, 300 Hz or 1000 Hz to the inputs DMIC1L and DMIC1R, you can see that the algorithm switches between the presets.

The APS Algorithm selects preset 1 with the best noise cancellation at 30 Hz if you feed a sine wave with 30 Hz into the DMIC1 inputs because according to the simulation the best results of cancellation are reached with this filter. The same you can also be observed with the other presets.

11 ANC Characterization

In order to provide all the necessary input for the FleX Project, Characterization measurements of the headphones / earbuds have to be done.

For basic ANC headphones this is done using either an in-ear coupler like the 711 coupler or an artificial head, like the GRAS 45.

To ensure a good measurement outcome, following points have to be controlled:

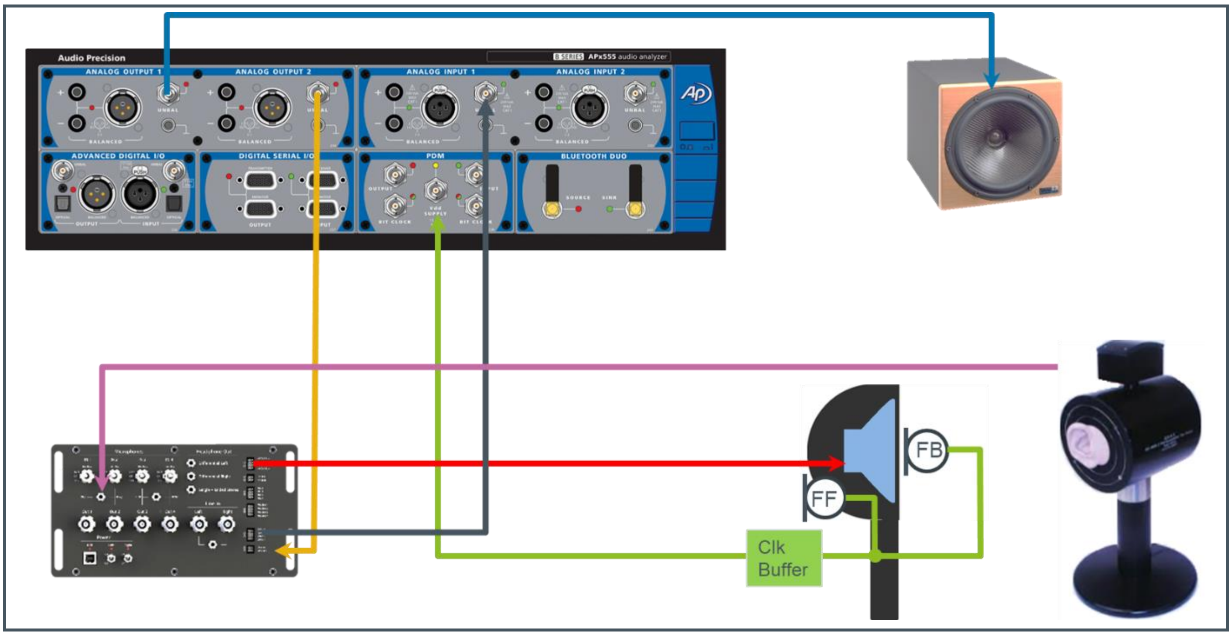
- The earbud has to have a good sealing in the coupler
- Leakage during measurement has to be avoided
- A good and tight fit of the headphone on the artificial head has to be ensured

11.1.1 Measurement

The measurement can be performed with the Audio-Precison Templates "Hybrid_Characterization.approjx" or "Hybrid_Characterization_FB_FF_seperate.approjx", which will be provided by **ams**.

The measurement setup looks as follows:

Figure 124:
ANC Characterization Measurement Setup



The Audio-Precision Templates provide the following output files:

- D2E
- D2FB
- D2FF
- A2E
- A2FB
- A2FF
- A2E (Open) → remove headphone/ earbud from measurement equipment

These output files can directly be loaded into the FleX Project.

12 ALC Characterization

The characterization process for an ALC headphone is slightly more complex compared to standard ANC, since characterization has to be done for multiple different acoustic leakage conditions. Acoustic leakage describes the fit of the earbud inside the listener's ear. If the earbud sits tightly inside the ear, the leakage is very low and if it sits loosely inside the ear then the leakage is high. The ALC algorithm running on AS3460 will adapt the ANC filters in real time to compensate for this leakage to give constant ANC performance for all acoustic leakage conditions.

It is necessary to characterize 5 different leakage conditions to cover the whole acoustic leakage range that can occur when people wear the headphones.

12.1 Characterization Methods

There are two different methods to do the characterization.

Method 1: Characterization is done on the Leaky Measurement Adapter (LMU)
Method 2: Characterization is done on the human ear

12.1.1 Method 1: LMU

For In-ear type earbuds (leaky and silicone ear-tip type) **ams** has developed the LMU, which allows characterization under different acoustic leakage conditions. The amount of leakage can be adjusted by the user by placing spacers of different thickness (1) in between the earbud holder (2) and the rest of the adapter. The LMU was optimized to behave very similar to a real human ear acoustically.

Figure 125:
Leaky Measurement Adapter (LMU) Measurement

There are two different holders available, one for leaky type In-ear headphone (see below) and one for silicone ear-tip type headphone that is very similar to a 711 coupler. When using the leaky type holder make sure to seal any leaks between the earbud and the holder with acoustic putty such that acoustic leakage is only introduced by the leakage spacer.

The “AP_ALC_Template_UCU.approx” AP template provided by **ams** should be used here. The template will generate the following output files: D2E, D2FB, D2FF, A2E, A2FB, A2FF, A2E(open). The characterization process should be repeated for the following 5 different leakages.

In-ear earbud (leaky):
In-ear earbud (silicone eartip):

The output files of all 5 leakage measurements have to be loaded into the corresponding sections in FleX.

Figure 126:
Load Characterization Data into FleX

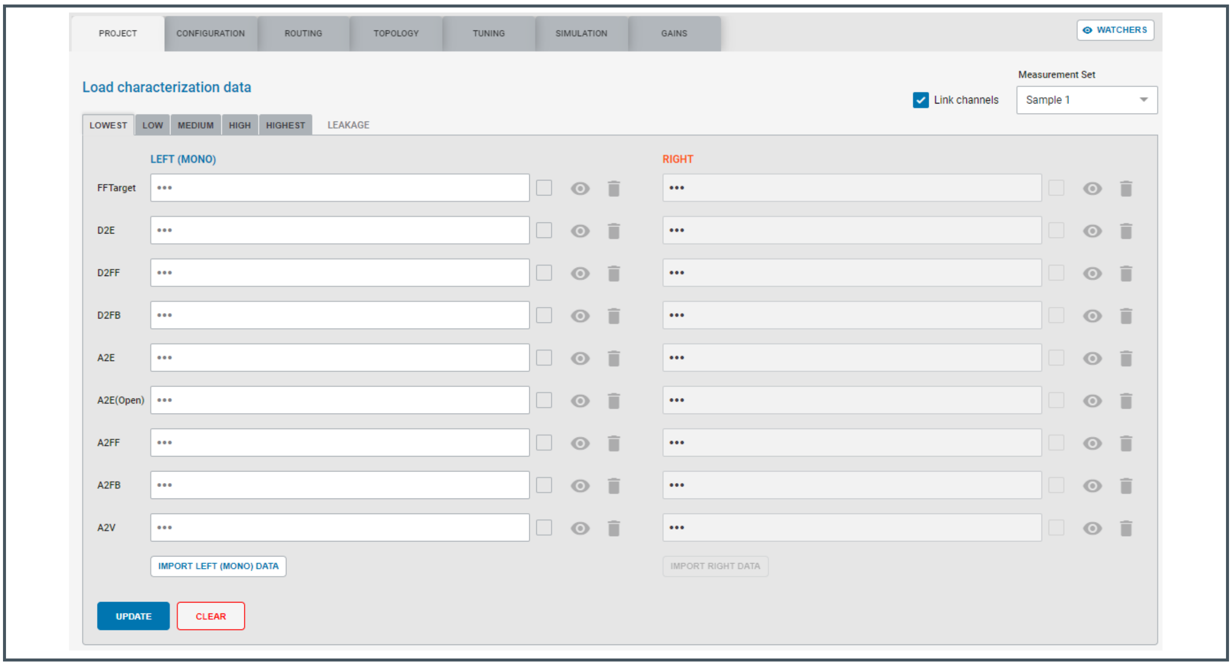
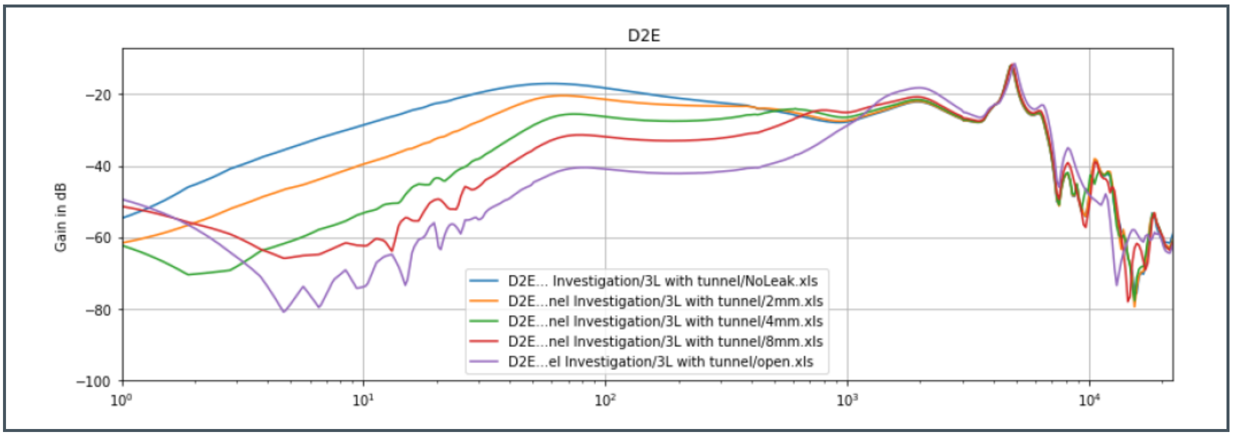
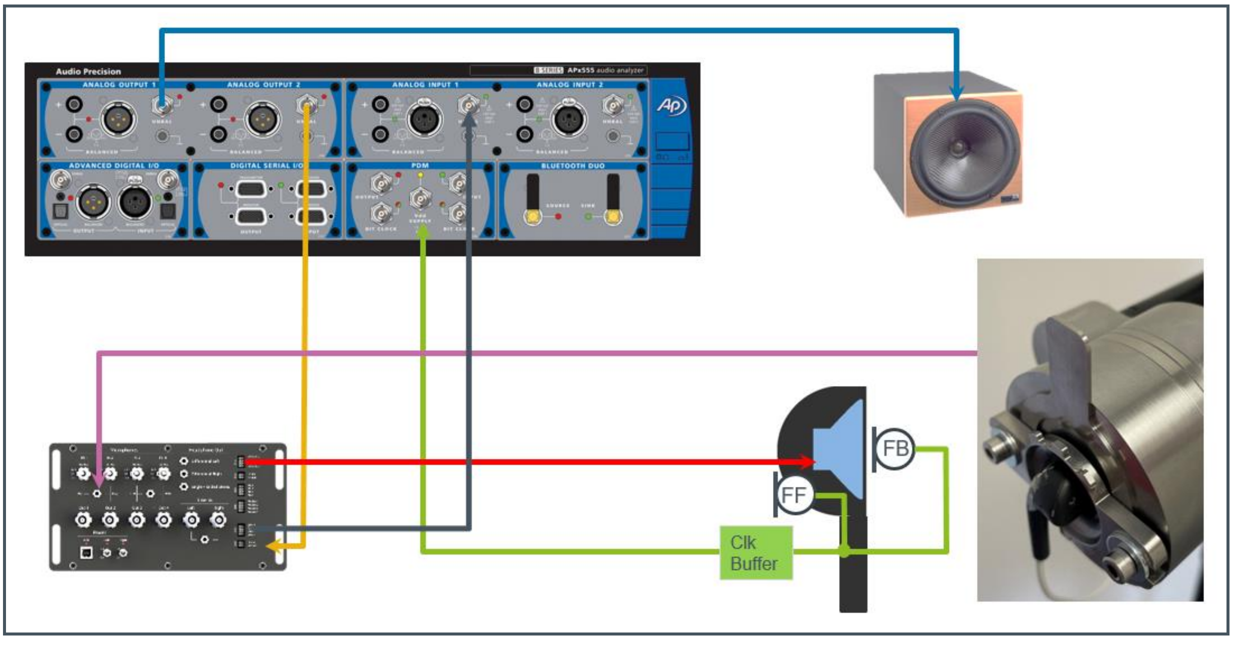


**Figure 127:
D2E Measurement for Different Acoustic Leakages**



The figure above shows the D2E response for different leakages. It can be seen that the bass response decreases with increasing leakage. The range in between the lowest and highest leakage is the leakage range that the ALC can compensate. The highest leakage that can still be compensated is physically limited by the acoustics of the headphone and the speaker that is used. As a rule of thumb the difference between lowest and highest leakage should not be more than 25 dB at around 100 Hz. For In-ear headphones with silicone ear-tips this difference should be slightly lower (around 15 to 20 dB) since typically smaller speakers are used compared to leaky type In-ear headphones.

**Figure 128:
LMA Measurement Setup**



The headphone characterization front end (black box lower left corner) is used to drive the headphone speaker (any analog linear headphone amp will do) and as a preamp for the reference mic signal coming from the LMA. The ambient speaker (upper right corner) should be placed on the same height as the headphone in approx. 0.5 m to 1 m distance.

12.1.2 Method 2: Human Head

The characterization for the 5 different acoustic leakages can also be done directly on the human ear. For the lowest leak condition the fit inside the ear should be very tight and well sealed and for the highest leak the fit should be loose but not too loose (D2E approx. 20 dB below lowest leak, see previous section). The intermediate leakages should be evenly spaced between lowest and highest leak. The D2E response is a good indicator for the amount of leakage that is present.

Figure 129:
Human Head Measurement

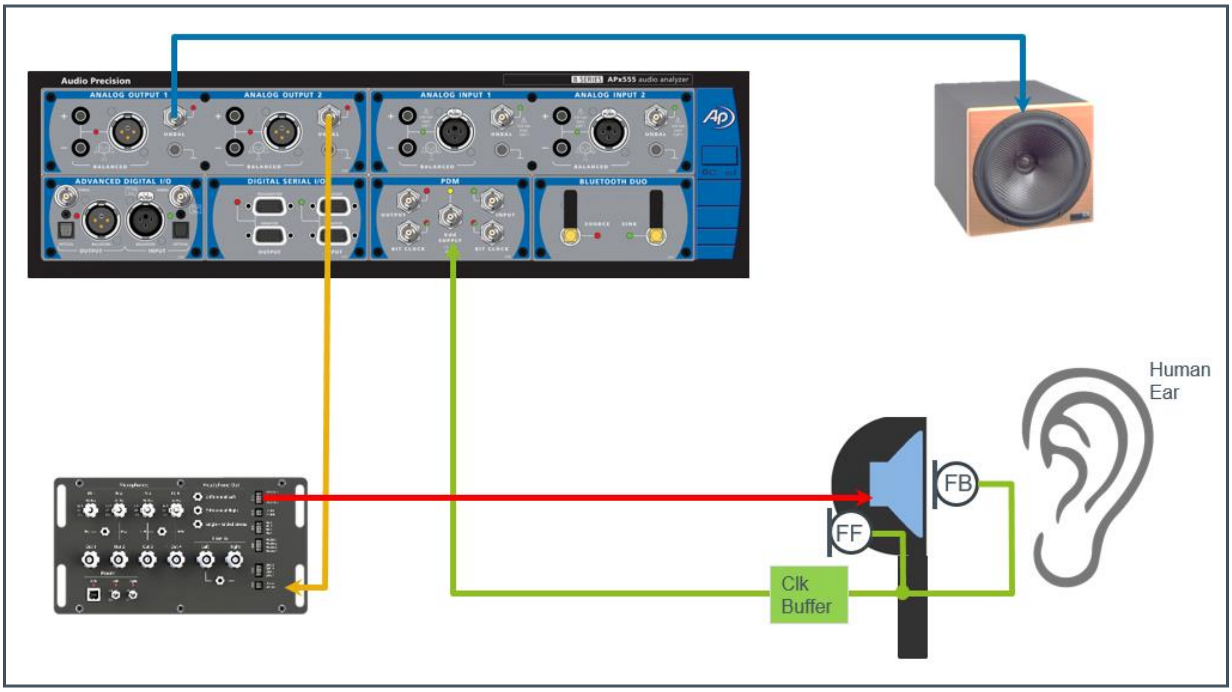
It is very difficult to insert a reference microphone into the human ear, which means that the FB microphone has to be used instead.

Output Files

- D2FB → used in FleX "D2FB" and "D2E"
- D2FF → used in FleX "D2FF"
- A2FB → used in FleX "A2FB" and "A2E"
- A2FF → used in FleX "A2FF" and "A2E (Open)"

The "AP_ALC_Template_Human_Ear.approx" AP template provided by **ams** should be used here.

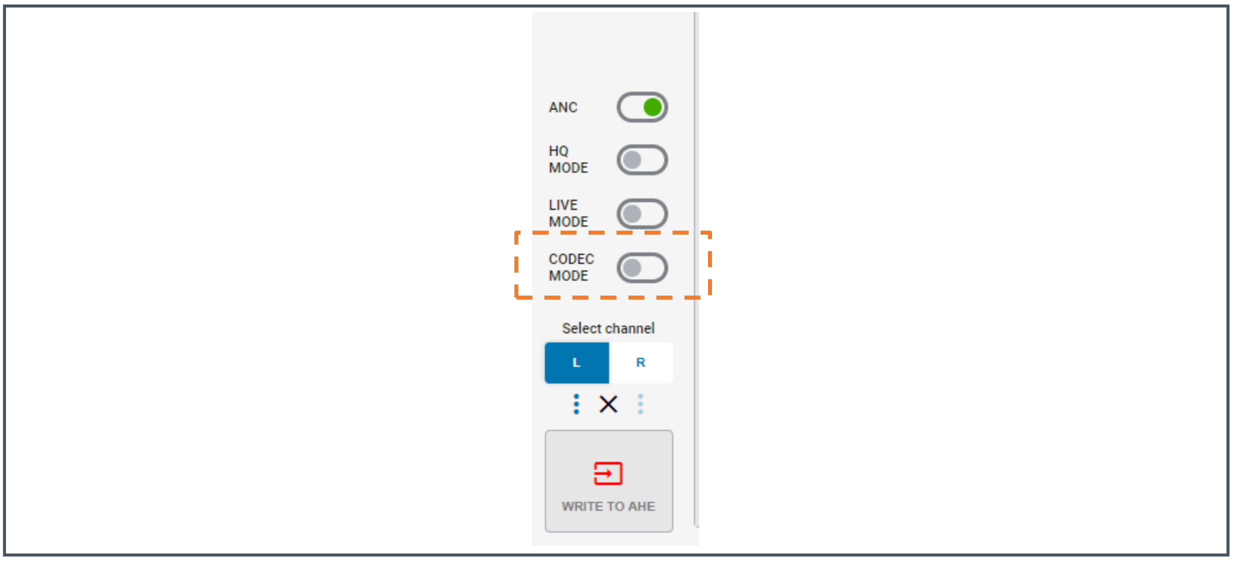
Figure 130:
Human Head Measurement Setup



13 Codec Mode

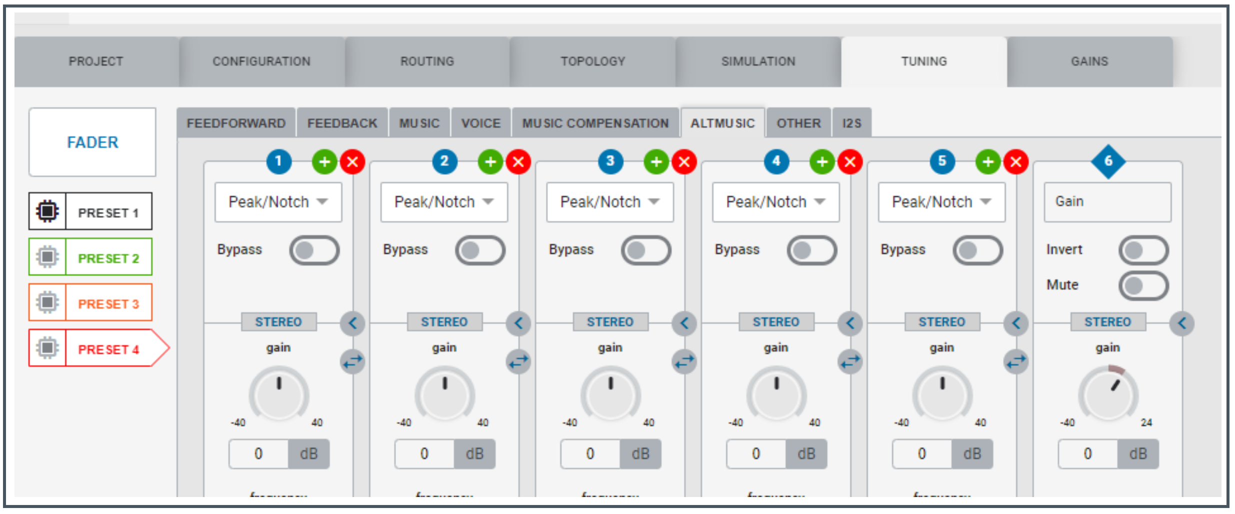
Codec Mode is a power saving music only mode (ANC Off).

Figure 131:
Codec Mode Enabling



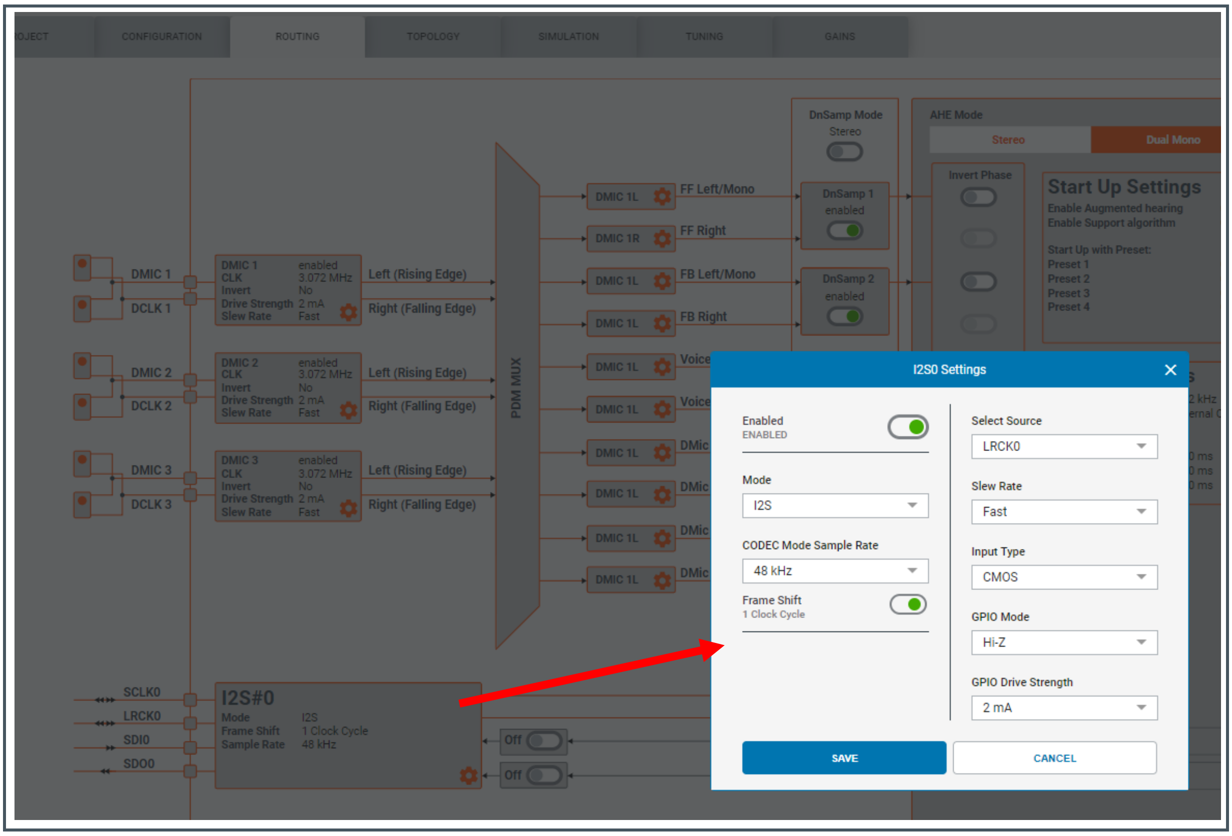
The music filters for Codec mode must be applied in the ALTMUSIC tab.

Figure 132:
Altmusic Tab



A valid clock signal on the I^{2}S port is mandatory to run the codec mode. The proper sample rate must be adjusted in the I^{2}S settings on the routing tab.

Figure 133:
I2S0 Settings



As long as AS3460 is in Codec operation mode a host MCU is not supposed to write to any other register than the AUDIO_CONTROL register. Please refer to the data sheet for more details.

14 Deep Sleep Mode

The Deep Sleep operation mode is a special mode that can be used to utilize the headphone amplifier of a Bluetooth SoC (System on Chip) device while augmented hearing is disabled. This implementation might bring power advantages in case the SoC device features a headphone amplifier with higher efficiency than AS3460.

To enable the deep sleep mode, sequentially write 0xa and 0x7 to the following register:

Figure 134:
Deep Sleep Mode Register

0xc		0x0000010b	
Bit	BitName	Value	Access
31 : 28	enter_deep_sleep	0x0	RW

Please refer to the AS3460 data sheet for more details.

15 Wind Noise Detection

Figure 135:
Wind Noise Detection Parameters in FleX

15.1 Overview

This section describes the functionality of the wind noise suppression and the process to set its parameters for which will likely require tuning for each new demonstrator / headphone model.

Pre-Requisites

- In order to simulate environments with different levels of wind noise consistently, a variable-speed electric fan is recommended.
- The wind noise suppression will work for ANC and ALC topologies
- FleX should have downsampler 3 configured with an additional microphone so that the wind noise detector can use the difference between the two external microphones to estimate wind noise.
- All other filter design and tuning should have taken place already, so that the amount of suppression that is required to eliminate obviously audible wind noise can be accurately determined.
- In FleX, under the tabs **Configuration -> Wind Noise Detection,** the wind noise detection must be enabled in order to refresh the latest wind level estimate in the **WIND_NOISE_METRIC** runtime register. In ALC-based topologies, the support algorithm must also be enabled under the **Configuration -> ALC** tab.

N.B. A version of FleX that supports a firmware version greater or equal to v1.22.0 will be required in order to access the WIND_NOISE_METRIC register. In order to use this with a static ANC based topology, a firmware version greater or equal to v1.23.4 will be required.

15.2 Operation

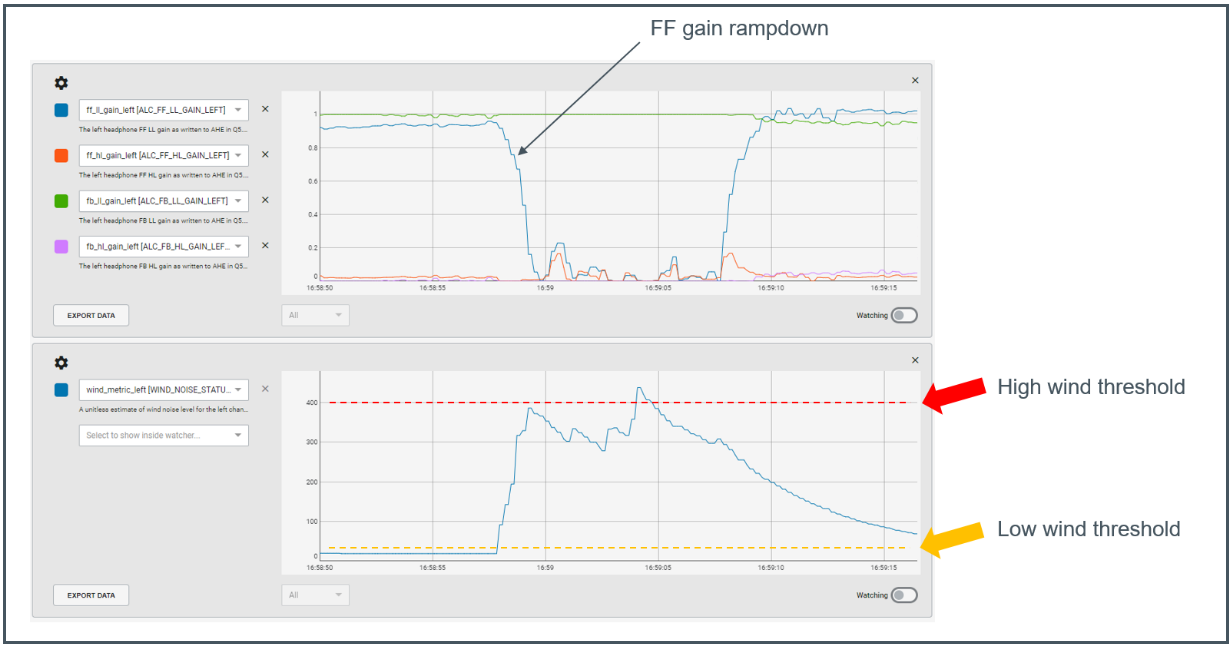
Wind can create very loud sound pressure at the outer microphone positions of the headphone (FF & Speech microphone). This can lead to digital signal clipping inside the AHE engine and audible distortions. To prevent these audible distortion artifacts the wind noise suppression uses the two outside microphones to detect wind. Once wind is detected the FF and Voice microphone gains will be ramped down to avoid clipping. The FB gain will stay untouched since the FB microphone is positioned inside the headphone and thus not exposed to wind.

From the two outside microphones a wind metric is permanently calculated that describes the probability of wind events. The wind metric value can be observed by using a watcher in FleX. Two thresholds can be set to determine at which wind metric level the wind suppression will activate and at which level it will fully ramp down the FF and Voice gains to the maximum suppression. These are the “low wind threshold” and the “high wind threshold” and the maximum suppression level is set by “Preset X fully suppressed gain”, which can be individually set for each preset.

For example:

If Low Wind Threshold = 300, then when the wind metric is lower than 300 then there is no suppression (0%). If High Wind Threshold = 600, then when the wind metric is greater than 600 the suppression applied is the maximum configured (100%) for the preset and voice paths. The amount of suppression varies linearly between the two above thresholds, so at a wind metric level of 450, 50% of the configured suppression is applied.

Figure 136:
Wind Noise Detection Parameters in FleX



In the ALC example shown in the figure above it can be seen how the FF gains ramp down once the wind metric exceeds the low wind threshold. It can also be seen that the FB gains are not affected.

15.3 General Tuning Process

Under normal circumstances, only the first two steps in this section need to be followed so that the activation range can be adjusted. The optional extra steps for range adjustment and metric fine-tuning are available to help debug specific use-cases that may differ from the default setup.

Pre-Tuning Setup

- Calculate the length of the acoustic path between the two feedforward microphones and set this in the **"External Mic. Separation"** field. At the time of writing, the supported sample rates are 96 kHz and 192 kHz, with distances in the range 5 mm - 50 mm. (For best results, this distance needs to be measured with an accuracy of ± 0.8mm if operating at 192 kHz or ±1.6 mm if operating at 96 kHz.)
- ***Wind-noise detection (and for ALC topologies, the support algorithm) should be configured to be on by default***. After any write to the AHE, ensure a few seconds are allowed before evaluating any changes. This will allow the ARM to update the required configuration parameters that the wind noise suppression metric requires.

Activation Range

"Low Wind Threshold"

Setup an environment where a low level of constant wind is present around the earphone. The level of wind is subjective to the user, but it is recommended to be set to a level where minimal audio artefacts are audible when the earphone is running in a medium leak configuration. Set the low wind threshold to a value just above the peaks being indicated in the runtime register **WIND_NOISE_METRIC**. This establishes a baseline for when noise suppression will start to activate on the feed-forward path when the wind increases above this current level.

"High Wind Threshold"

Setup an environment where the level of high constant wind around the earphone would normally cause lots of noise on the feedforward path, indicative of an environment where it should be fully suppressed. Set the high wind threshold to the average of the value being indicated in the runtime register **WIND_NOISE_METRIC**. This establishes the worst-case expected environment which will activate maximum feed-forward gain suppression. Any increase in wind about this level will have no further effect.

Fine Tuning

This step can be skipped unless there are specific use-case problems that altering the metric response time behaviour may address.

“Wind Detection Attack Time”

This governs the attack time of the wind noise metric. It is recommended to keep it low for quick responses. This does not control the gain adjustment rate, just the metric ramp up time of **WIND_NOISE_METRIC** in the presence of possible wind.

“Wind Detection Decay Time”

This governs the decay time of the wind noise metric. It is recommended to keep it much longer in comparison to the attack time, to help smooth out the overall response and subsequent mitigation actions. This does not control the gain adjustment rate, just the metric ramp down time of **WIND_NOISE_METRIC** in the absence of possible wind.

“Gain Suppression Rate”

This sets the maximum rate at which gain suppression can take place (i.e. if the wind metric instantly changed from 0 to requiring maximum wind suppression). It is effectively a speed limit, recursively applied to the current gain at a rate of 12 kHz. Decrease the value for faster attenuation, or increase towards 1.0 for slower attenuation. A value of 1.0 provides no attenuation. Note that this only affects how quickly the gain will respond to rapid increases in the current wind level, as observed in runtime register **WIND_NOISE_METRIC**.

15.4 Parameters

Low/High Wind Threshold:
Once the wind metric is above the low threshold the wind noise suppression will activate and cap the FB/FF gains to a lower value. When the wind metric reaches the high threshold then this cap value is the max suppression value defined by the user.

Attack/Release Time:
These are applied to the wind metric. Longer times will smoothen transitions.

Gain Suppression Rate:
Is the factor applied to the FB/FF gains during the transition from current gain value to the capped target value. Smaller means faster ramping.

Wind Detection Noise Floor:
FF signal energy is checked, if it is below this value then the wind metric is set to 0 to avoid false positives.

Wind Detection Sensitivity:
Scaling factor for the wind metric. A value of 200% will multiply the wind metric by a factor of 2.

Preset X Fully Suppressed Gain:
Maximum gain reduction that will be applied when the wind metric exceeds the high wind threshold. This parameter can be set individually for each preset. A value of 0 means that the FF/Voice gains will be muted, a value of 0.5 means that the FF/Voice gains will be attenuated by -6dB.

16 Revision Information

Changes from previous version to current revision v4-00	Page
Clarification of I2S address section using GPIO_A0 at boot	25
Corrected capitalization of ‘FleX’ throughout the document	1-124

- Page and figure numbers for the previous version may differ from page and figure numbers in the current revision.
- Correction of typographical errors is not explicitly mentioned.

17 Legal Information

RoHS Compliant & ams Green Statement

RoHS Compliant: The term RoHS compliant means that ams AG products fully comply with current RoHS directives. Our semiconductor products do not contain any chemicals for all 6 substance categories plus additional 4 substance categories (per amendment EU 2015/863), including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, RoHS compliant products are suitable for use in specified lead-free processes.

ams Green (RoHS compliant and no Sb/Br/Cl): ams Green defines that in addition to RoHS compliance, our products are free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material) and do not contain Chlorine (Cl not exceed 0.1% by weight in homogeneous material).

Important Information: The information provided in this statement represents ams AG knowledge and belief as of the date that it is provided. ams AG bases its knowledge and belief on information provided by third parties, and makes no representation or warranty as to the accuracy of such information. Efforts are underway to better integrate information from third parties. ams AG has taken and continues to take reasonable steps to provide representative and accurate information but may not have conducted destructive testing or chemical analysis on incoming materials and chemicals. ams AG and ams AG suppliers consider certain information to be proprietary, and thus CAS numbers and other limited information may not be available for release.

Headquarters
ams AG
Tobelbader Strasse 30
8141 Premstaetten
Austria, Europe
Tel: +43 (0) 3136 500 0

Please visit our website at www.ams.com
Buy our products or get free samples online at www.ams.com/Products
Technical Support is available at www.ams.com/Technical-Support
Provide feedback about this document at www.ams.com/Document-Feedback
For sales offices, distributors and representatives go to www.ams.com/Contact
For further information and requests, e-mail us at ams_sales@ams.com