

Echo AIO™ Test Interfaces



4.3 Software for macOS

Introduction

This disk image contains all the software necessary to use Echo AIO Test Interfaces with an Intel or Apple Silicon Mac:

- The AIO Control Panel is a graphical user interface application for configuring your AIO. The AIO Control Panel provides full control over the hardware, shows peak metering for all channels, and can save and load configurations.
- The EchoAIO CLI is a command-line app that can configure an AIO directly from the Terminal or by launching it from another application.
- EchoAIOInterface.dylib is a standard dynamic library providing a C-language interface for the AIO. The library is required for the EchoAIO CLI, or can be used to develop your own control application.

To install the AIO Control Panel, just drag-and-drop the icon into the Applications folder. The Echo AIO uses the built-in macOS USB audio driver; no driver installation is necessary.

For more information about the AIO Control Panel, please refer to the Echo AIO User Manual:

<https://echotm.com/library/echo-aio-user-manual/>

Version history

4.3

Interface module firmware

- Fixed channel names for S and H modules
- Support for more module configurations
- Fixed issues with AIO-Sv2 and AIO-Lv2 modules initialization
- Improved RMS metering
- Added enable/disable control for output limiter
- Automatically save and restore the audio clock source

Control panel application

- Fixed C module AUX IN not updating
- More flexible control layout (vertical channel stack if the window is narrow and tall)
- Streamlined meter controls

CLI + API

- Complete rework of API + CLI for new command syntax and support for all AIO modules

4.1

Interface module firmware

- Automatically load T module firmware
- Automatically save and restore T module configuration
- Fixed issue with channel labeling for AT & LT configurations
- Fixed startup issue with AT & LT configurations
- Added support for T1 configuration
- Added support for larger future firmware images
- Output meters are now calculated before the output calibration scaling

AIO-T module firmware

- Major firmware rewrite for reliability and timing fixes

Control panel application

- dBFS meters in RMS mode now read 0 dB for full-scale sine output
- Fixed channel naming for units with empty module slots

- More user-friendly display of firmware versions
- Fixed restoring the window size
- Fixed restoring meter modes & units
- Fixed TDM incorrect bits per word configuration
- Fixed channel layout for AIO-AT and AIO-LT configurations
- Changed AIO-S and AIO-H current monitor meters to read in milliamperes instead of ampere
- Added decibel ampere meters for AIO-S and AIO-H current monitor

4.0.0

Initial release of integrated installer with updated Control Panel and firmware

Echo AIO Command-Line Interface

Setup

The AIO CLI application enables you to configure your AIO from the macOS Terminal prompt or from another application. This is especially useful if you want to automate setting up your AIO as part of your test sequence.

To set up the CLI on macOS, mount the Echo AIO disk image and copy EchoAIO and EchoAIOInterface.dylib to a folder on your hard driver; for example, ~/aio.

The CLI application is named EchoAIO. To run the CLI, EchoAIOInterface.dylib needs to be in the same folder as EchoAIO.

Configuring a single channel

To configure a single channel on your AIO, navigate to the CLI folder using the Terminal.

Specify the channel configuration with the command line:

```
EchoAIO input=<channel> [gain[={1|10|100}]] [ccp[={on|off}]] [teds]  
EchoAIO output=<channel> [gain[={1|10}]]
```

Parameter	Description
input	Specify input channel starting at channel 1
output	Specify output channel starting at channel 1
gain	Print the current gain setting
gain={1 10 100}	Set the channel to 1x, 10x, or 100x gain
ccp	Print the constant current power setting
ccp={on off}	Enable or disable constant current power
teds	Read TEDS data from the specified input and print out the data

Parameters are not case-sensitive.

For example:

To set 10x gain and enable the constant current for MIC3 for input 3:

```
cd ~/aio  
./EchoAIO input=3 gain=10 ccp=on
```

To print out the TEDS data and current settings for MIC1 on input 1:

```
./EchoAIO input=1 gain ccp teds
```

Configuring the entire unit

You can configure the Echo AIO unit using the CLI with Control Panel session files. To create a session file, run the AIO Control Panel app, set up the AIO unit as required, then select File/Save.

To configure the unit with a session file, navigate to the folder containing the CLI using the Terminal. Then, enter the command:

EchoAIO file= <Session file>

For example:

To load a file called session.AIO-A1 enter:

```
./EchoAIO file=session.AIO-A1
```

API Library

If you would like to develop your own application to control your AIO, please refer to this repository for API documentation along with C++ and Python examples:

<https://github.com/mattgonzalez/EchoAIOExample>