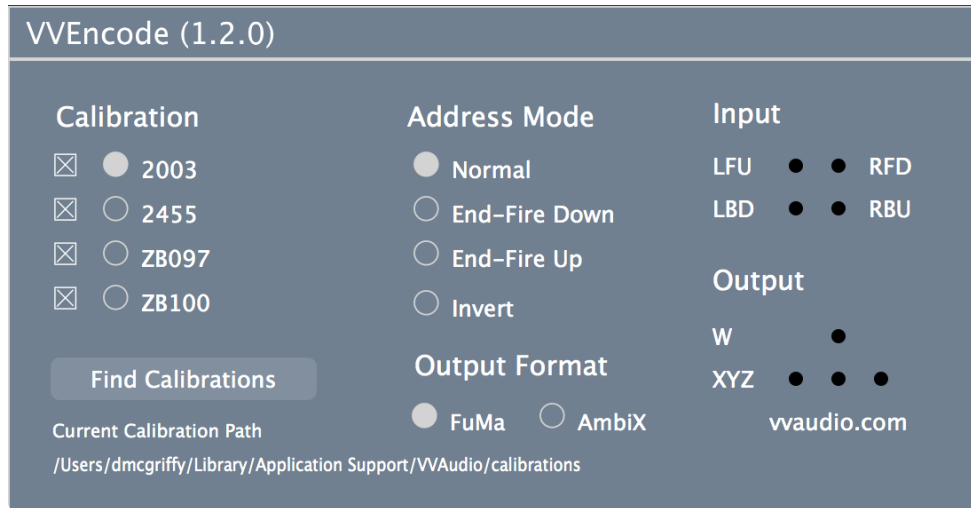


# Getting Started with VVEncode

---



This document will help you get started using the VVEncode plugin from VVAudio. VVEncode takes signals from a tetrahedral microphone (called A-format) and converts them into B-format using the calibration files supplied by the manufacturer. Both the TetraMic and Brahma are supported, as well as DIY microphones like the Ambi-Alice.

To get the most out of this document, you should have some basic familiarity with the process of recording using computers. You need not be an expert in either digital recording or ambisonics.

We will cover the installation, setup and use of VVEncode. With this, you should be able to use VVEncode to convert your microphone signals into B-format. A complete ambisonic workflow will also include at least a decoder and perhaps some processing. We will see how VVEncode fits into this larger picture.

## Table of Contents

<b>Installation</b> .....	<b>3</b>
<b>MacOS</b> .....	<b>3</b>
<b>Windows</b> .....	<b>3</b>
<b>Loading Calibration Files</b> .....	<b>3</b>
<b>Setting Address Mode</b> .....	<b>4</b>
<b>Activity Lights</b> .....	<b>4</b>
<b>Placement</b> .....	<b>5</b>
<b>Routing</b> .....	<b>5</b>
<b>Single 4-channel file</b> .....	<b>5</b>
<b>Four mono files</b> .....	<b>6</b>
<b>Ambisonic Workflows</b> .....	<b>8</b>
<b>Field recording for stereo</b> .....	<b>8</b>
<b>Field recording for surround</b> .....	<b>8</b>
<b>Mixing with Ambisonics</b> .....	<b>8</b>

## Installation

Before installing VVEncode, you should have downloaded the appropriate plugin package for your operating system and the calibration files for your microphone. After that it all depends on which platform you are using.

### MacOS

The MacOS versions of VVEncode are distributed as a .pkg file that will install both VST and AAX to the correct locations. Note that you will need to right-click on the .pkg file and select “open” instead of simply double clicking on it in order to satisfy Apples security policies.

### Windows

In Windows, VST’s can go anywhere and most VST hosts have some way of choosing the VST directory. The default location for 64-bit plugins, as defined by Steinberg is “C:\Program Files\Common Files\VST2”.

The Windows version of VVEncode is distributed as an installer package. This installer puts the AAX plugins in their proper location and allows the user to choose the location for the VSTs.

## Loading Calibration Files



Both the TetraMic and Brahma are supplied with individual calibrations files. If you have previously been using VVTetraVST or VVMic, then VVEncode will find the files where they are. If have not previously used your calibration files, then do the following:

1. Download the calibration files to your local disk
2. Unzip them if needed
3. Start a host program like Reaper or ProTools
4. Create a four channel track and place VVEncode on it (see placement below).
5. Press “Find Calibrations” and navigate to the place where you downloaded your files
6. Select either the .txt or .tetra file in the case of a TetraMic or the folder containing the .wav files in the case of a Brahma and give this calibration a short name

The calibration files you just loaded should now show up in VVEncode using the name you gave them. You can delete them by pressing the ‘x’ next the name. Delete and re-find to change the name. You can have several calibrations loaded at once.

VVEncode actually searches several places for calibration files, depending on operating system:

Windows:

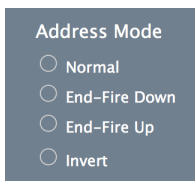
```
Documents\VVAudio\calibrations
C:\Program Files\Common Files\VVAudio\calibrations
C:\Program Files (x86)\Common Files\VVAudio\calibrations
C:\Program Files\VVAudio\VVMic\calibrations
C:\Program Files (x86)\VVAudio\VVMic\calibrations
```

OSX:

```
~/Library/Application Support/VVAudio/calibrations
/Library/Application Support/VVAudio/calibrations
~/Library/Audio/Plug-Ins/VST/calibrations
/Library/Audio/Plug-Ins/VST/calibrations
```

You can see where VVEncode found the calibration files it is using down at the bottom of the plugin's window.

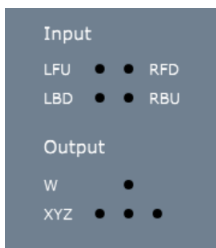
## Setting Address Mode



Address mode refers to which side of the mic is facing the front. With tetrahedral microphones, the whole idea is to make it as close a possible to perfectly symmetrical, but sometimes it is more convenient to have the handle of the mic facing up or back. This chart explains the different modes:

Mode	TetraMic	Brahma
<b>Normal</b>	Handle down, logo forward	plug down, serial # forward
<b>End-Fire Down</b>	Handle back, logo down	plug back, serial # down
<b>End-Fire Up</b>	Handle back, logo up	plug back, serial # up
<b>Invert</b>	Handle up, logo forward	plug up, serial # forward

## Activity Lights



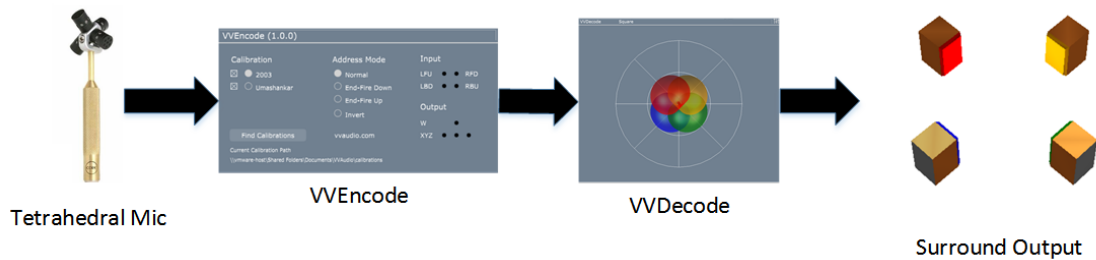
The activity lights show when there is signal at the input and output. They start out black, then as the signal gets louder, they get brighter green, then fade to yellow and finally turn red at -1 dBFS. The main point of the input light is to make sure all four channels are getting to the plugin and not, for example, getting mixed into the first two. The output lights mostly show when clipping might occur. Note that most tetrahedral mic calibrations add considerable bass so the output will often be louder than the input.

## Placement

VVEncode must be placed on the right kind of track to be useful. In Reaper, this can be any track with four or more channels. In ProTools, this can be either a Quad track or a 1st order ambisonics track. VVEncode will nearly always be the first plugin in the track since most types of processing will disturb the spatial info. Changing the gain identically on all four channels is acceptable.

## Routing

Setting up the routing in your particular host software varies considerably depending on the host. The thing to remember is that all four mic signals must get to the same copy of VVEncode. You should not use multiple copies of VVEncode unless you are mixing multiple tetrahedral mics. It can be as simple as this:

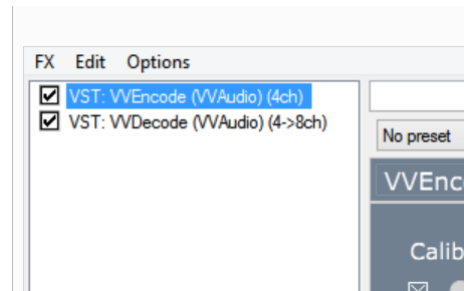


## Single 4-channel file

If you have recorded to a single, four channel file, then the routing and plugin placement is quite simple, at least as far as VVEncode. Create a four-channel track for the file and insert VVEncode as the first plugin on it. In ProTools this should be a "1st Order Ambisonic" track. You can put other ambisonic plugins after this, as discussed below in Routing, or mix the resulting B-format with other ambisonic sources.



ProTools



Reaper

### Four mono files

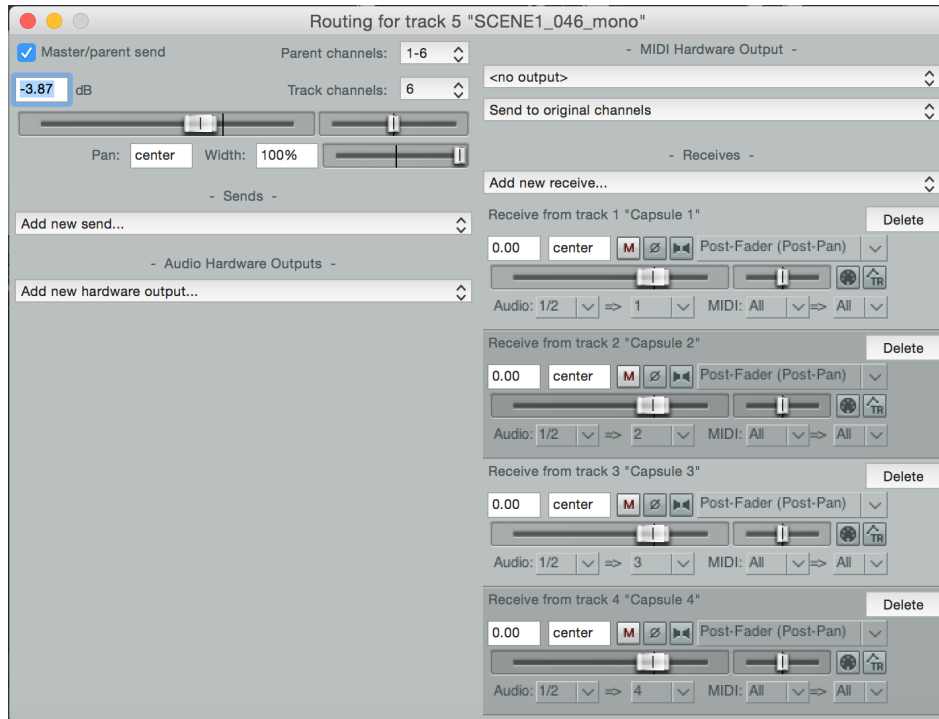
If you have four mono files, place each one on a separate track. Then each of these four tracks must be routed to one of the four channels of a fifth track. VVEncode is placed on this fifth track. To see how this works in various hosts, look at the templates included with the plugin download. In ProTools you must create a quad bus with four mono sub busses.

A screenshot of a ProTools Quad bus configuration. The interface shows a dark grey background with a table of bus settings. The table has columns for '1stOrdrAm', '00', '01', '02', and '03'. The rows are labeled 'B-Format 1' through 'B-Format 4'. The '1stOrdrAm' column shows 'Mono' for all rows. The '00' column has 'M' for 'B-Format 1'. The '01' column has 'M' for 'B-Format 2'. The '02' column has 'M' for 'B-Format 3'. The '03' column has 'M' for 'B-Format 4'.

✓ ▾ B-Format	1stOrdrAm	00	01	02	03
B-Format 1	Mono	M			
B-Format 2	Mono		M		
B-Format 3	Mono			M	
B-Format 4	Mono				M

ProTools Quad bus with sub busses

In Reaper, you can route the mono tracks to the channels of the fifth track directly.



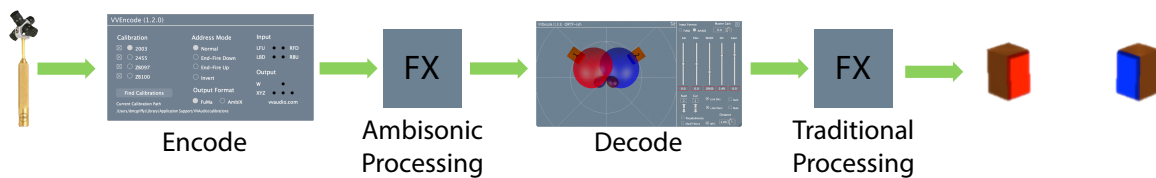
Reaper

## Ambisonic Workflows

Ambisonics can do lots of things. Here are a few common cases.

### Field recording for stereo

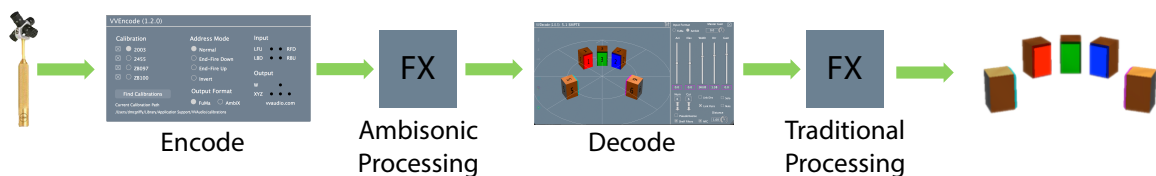
While people generally associate ambisonics with surround sound, it can be very effective at producing stereo content as well. Having captured the complete soundfield, you can then use a decoder to produce a pair of virtual microphone outputs in post production. This allows you to choose the direction and directivity of each microphone back in the studio. You can make the stereo spread wider or narrower to control the stereo image. You can choose from cardioid, hyper-cardioid, figure-8 or anything in between to control the amount of room sound.



Between the encoder and decoder you can use any of the increasing number of ambisonic specific effects available. After the decoder, you can use standard stereo effects like EQ or limiters just as you would with any stereo recording. Also note that while it is usual to archive the raw recording and the final master, with ambisonics it often makes sense to save the B-format result of the encoder. This will allow you to remaster the recording in the future without needing the calibration files for the specific microphone used.

### Field recording for surround

For basic surround recording with a single, tetrahedral microphone, the process is exactly as described above for stereo, except you choose a different preset on the decoder and surround aware plugins after the decoder.



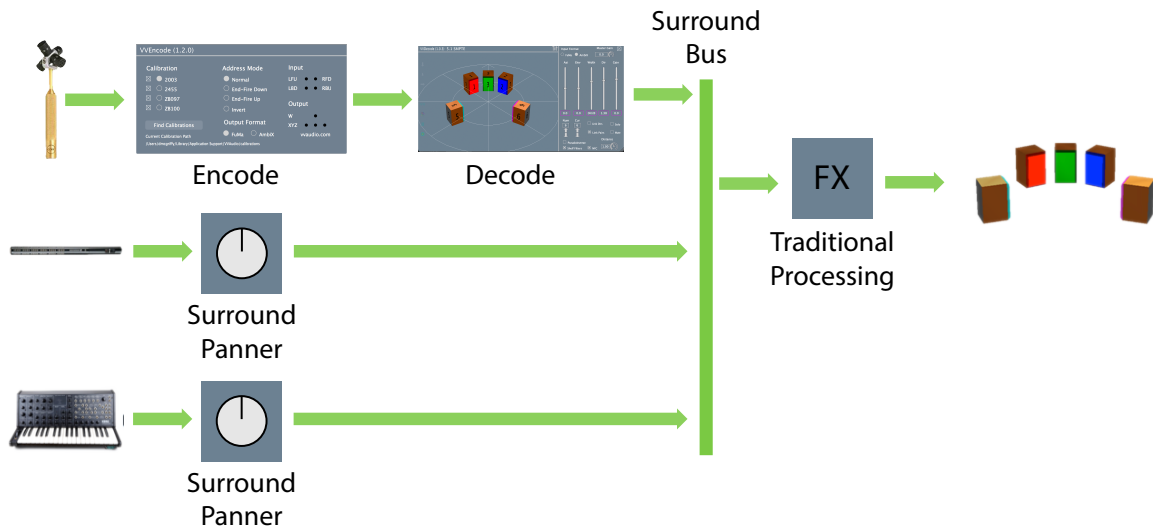
This ability to take the exact same session with all of its editing and processing and render it first to stereo and then to surround with the flick of a switch is part of the power of ambisonics. You could even send the same B-format signal to two decoders and render both at the same time.

### Mixing with Ambisonics

Most productions will use more than a single microphone, of course. You can combine ambisonic and other sources in a mix in a couple of ways. If your target format is stereo, 7.1.2 or any traditional format, then you place both the encoder and



the decoder in sequence on ambisonic tracks and set the decoder output to match



the format of your main bus. Other channels are mixed using standard panners. If your target format is ambisonic, perhaps for a VR production, then use an ambisonic main bus with ambisonic panners and encoders to mix your sources into this bus. Then route the main bus to the master channel or an aux track with any ambisonic processing you want and a decoder for monitoring. When it's time to export the results you can either disable the decoder or render from the bus prior the decoder. Here's how that looks:

