Eventide®

H9000

P/N: #141242

P/N: #141242
Version 2.2.11

Eventide Inc.
One Alsan Way
Little Ferry, NJ 07643
United States of America
201-641-1200

www.eventideaudio.com

Table Of Contents

2. Setup

3. Emote

4. Front Panel Operation

5. Hardware

6. Firmware

7. Expansion Cards

A. Startup Sequences

B. Signal Flow

C. MIDI Functionality

D. Specifications

E. Technical Support

F. Algorithm Development

G. Legalese

This page was intentionally left 99.88% blank.

1. INTRODUCTION

Congratulations on your purchase of the Eventide H9000, the new flagship processor in the Eventide product line. The H9000 is designed to be an incredibly versatile tool, allowing you to process a large number of channels using the latest digital and networked audio formats, and offering the great sounding effects you have always loved from Eventide.

The H9000 is available in two basic models:

- The H9000
- The H9000R

Although the H9000 front panel allows you to navigate the device with ease, we highly encourage you to download and use the full-featured control app/plug-in "Emote."

Note

Please visit https://etide.io/H9000-Manual to view the HTML version of this manual on your computer or smartphone. The HTML version includes the Algorithm Library, and Third Party Licenses, which are not present in the PDF or print version of this manual.

In this manual, when discussing features common to both units, we will refer to the hardware as "H9000".

1.1. What's in The Box

Your H9000 comes with the following:

- H9000 2RU rackmount processor
- Registration/Warranty card
- Wireless network (Wi-Fi) adapter
- FAT32-formatted USB drive
- USB cable
- Power cable
- Rackmount screws
- User's manual

1.2. Features

The H9000 is designed to be easy to use while giving you power, control, and versatility through the implementation of the following features:

Multiple Quad ARM Processors

The H9000 uses multiple cutting-edge effects processors, allowing it to process a large number of I/O channels and up to 16 separate effects algorithms simultaneously.

Flexible and Adaptable Effects Structure

Your H9000 provides up to four FX chains, with each chain containing up to four separate processing algorithms. FX chains can be routed in series, parallel, and more.

Large Complement of Analog and Digital I/O

Multiple formats and connector types allow you to integrate the H9000 into a variety of setups.

Expandable I/O

Three expansion card slots allow you to integrate your H9000 into the latest digital audio and networked environments. We are always developing new expansion cards, ensuring that your processor is as future-proof as possible as digital and networking formats evolve.

Full Front Panel Control

While we offer and recommend the Emote control app, the standard H9000 also provides intuitive front panel control using a large color display, context-sensitive soft keys, and a complement of physical navigation controls.

Network Connectivity

Connect to your H9000 with Emote via either your local area network, or a direct ethernet connection.

Computer Audio Interfacing

USB 2.0 allows you to interface your H9000 with your audio hardware and DAW as a standard audio interface with 16 channels of I/O. Send, process, and return 16 separate channels of individual DAW tracks.

Easy Sharing of FX Chains and Sessions

The H9000 uses FAT32-formatted USB thumb drives to save and import your custom effects settings, making it easy to back up, carry, and share your creative efforts.

Compatibility with VSig3 Editing Software

VSig3 is a cross-platform visual editor that enables the creation of custom algorithms for the H9000. Since VSig3 allows you to view some of Eventide's "secret sauce", Vsig3 is only available for H9000 owners, and requires H9000 registration. It is available for download here: https://etide.io/vsig3

1.3. Hierarchy

In order to get the most out of your H9000, please familiarize yourself with the four main terms below, which represent the "hierarchy" used in the product:

1.3.1. Session

A *Session* is the highest level on the H9000 hierarchy. Session presets save the entire state of the product at any given moment, including:

1. The currently loaded FX chains:
   - Parameters within the FX chains.
   - Routing assignments within the FX chains.
2. Routing information external to each FX chain, including:
   - Physical I/O of each FX chain container.
   - Direct I/O connections made between physical I/O.
   - Additional input/output gain settings for each I/O channel routed to/from the FX chain.
3. The current state of the global settings:
   - All the settings found in the Setup mode (these can be excluded by category when loading).

- All the Scene Maps found in Scenes mode, along with current active map index.
- LED meter mappings.
- The state of Global Mute/Bypass.
- Metadata (Name and ID are the most relevant to the user).

1.3.2. FX Chain

The H9000 allows you to chain up to four algorithms together into a custom *FX Chain*. The FX chain is fed by audio inputs, then the audio is routed through up to four algorithms, and then finally to audio outputs.

You are free to decide which algorithms are included in the FX chain, the order in which they process the audio, and even determine aspects such as separate parallel and series audio paths.

FX Chain presets save:

- Everything contained within each algorithm preset for all the algorithms present in the FX chain.
- General parameter values related to the entire FX chain container (mix, mute/bypass, input/output gains).
- Additional mappings of these FX chain parameters to MIDI controllers.
- All routing information internal to the FX chain (from the FX chain inputs, to/between algorithm containers, to the FX chain outputs). No physical I/O information is included.
- Function values, as well as mappings of the actual Function knobs to external controllers.
- Metadata (Name and ID are the most relevant to the user).

1.3.3. Algorithm

Over many years, Eventide has created a vast library of audio processing routines, which exist as effects processing “modules”. Some of these are simple, such as a low pass filter, a limiter, a compressor, etc., while others are more complex, such as an FFT or a reverb network.

An *Algorithm* in the H9000 is a combination of some of these modules, and of certain variable values associated with the modules that make up that algorithm. The designer of the algorithm might choose to set some variables of some modules to a specific value that cannot be changed by the user, while allowing that user to change the values of other variables.

The algorithms are the H9000's "secret sauce", with specific effects parameters that are adjustable by you, the user. The members of the H9000 Family each have well over one thousand five hundred algorithms, covering the whole range of audio effects.

Algorithm Presets save:

- Algorithm-related parameter values.
- General parameter values related to the algorithm container (mix, mute/bypass, input/output gains).
- Mappings of parameters to external controllers and Functions.
- Assorted metadata (Name and ID are the most relevant to the user).

1.3.4. Parameter

Parameters are the individual adjustments you can make within an algorithm (or FX chain). On the H9000, when you navigate to a specific algorithm you are presented with its specific parameters, which you can adjust.

A parameter might control one variable of one underlying processing module, or it may control several variables of several modules. Some parameters may be adjustable, whereas for others the designer may choose to set some of the underlying variables to a fixed value and not bring them out to the user of the effect.

For example, a reverb algorithm would present you with settings such as reverb time, room size, decay time, etc. In contrast, a guitar amp emulator might offer parameters such as gain, low/mid/high EQ, etc.

This page was intentionally left 99.88% blank.

2. SETUP

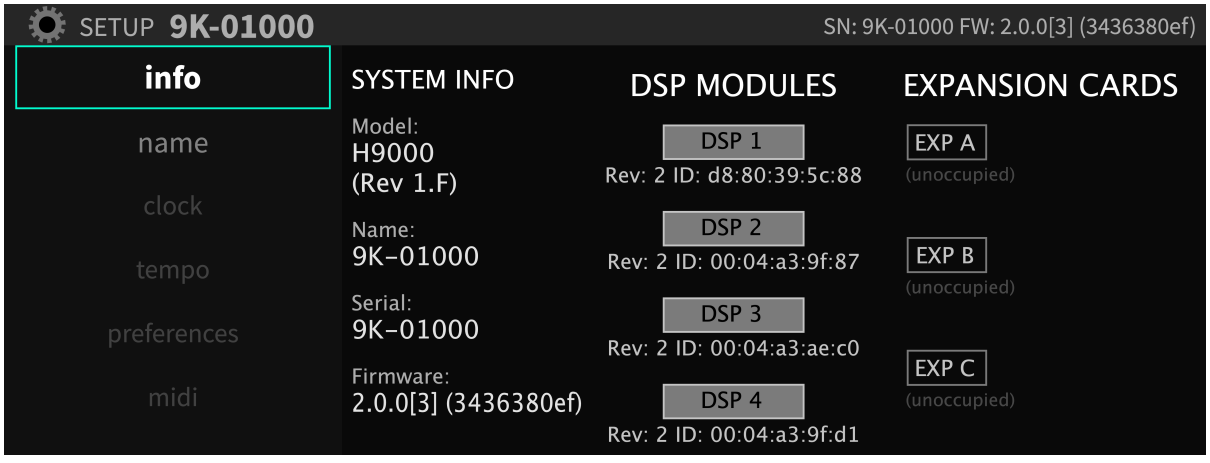
2.1. Setting up Your H9000

> **Note**
>
> Please refer to Section 3: Emote for instructions on setting up your H9000R.

Press the *Setup* button to enter the setup menu for the H9000. The left-hand side of the display shows a list of areas you can customize.

Use the cursor buttons or wheel to navigate to the setup screen you wish to adjust. The setup category will highlight and the right-hand side of the display will show the specific items that can be adjusted in that section.

2.1.1. Info



The *Info* screen is a display only; none of its settings can be adjusted. It provides a high-level overview of your H9000 unit, with the following information:

System Info

- Model: (In this case the H9000)
- Name: The custom name that you have assigned to your unit
- Serial Number
- Firmware Version

DSP Modules

This column displays revision and ID information for the individual hardware DSP modules installed in your unit. If a DSP module is displayed as (unoccupied) or OFFLINE, please refer to Section A.5: USB Self-Test Mode.

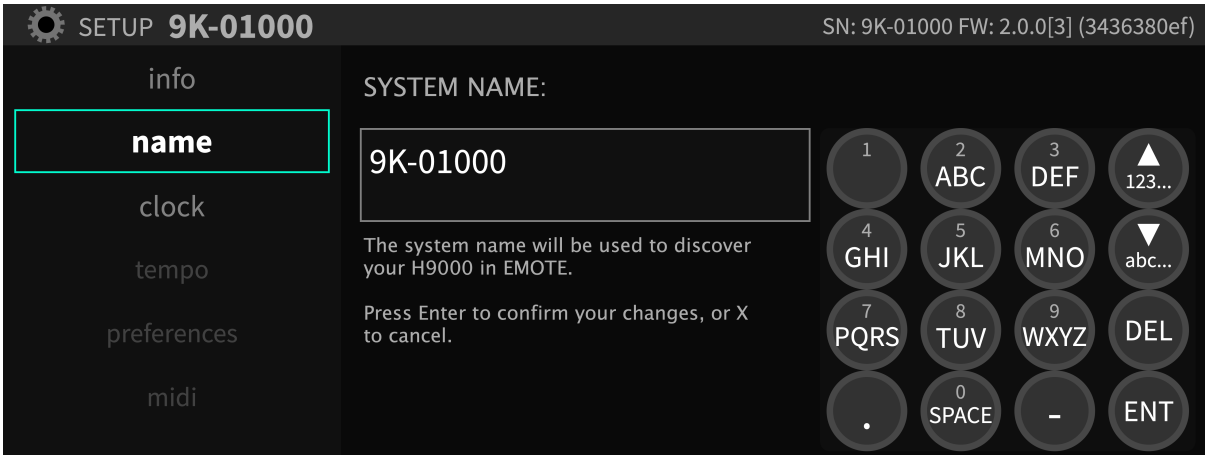
Expansion cards

This column displays information on any expansion card(s) that are installed in the H9000's three expansion slots (which are labeled slots A, B, and C).

2.1.2. Expansion Card Setup

This menu page displays any installed expansion cards and any relevant information pertaining to installed expansion cards (i.e. expansion card firmware version).

2.1.3. Name

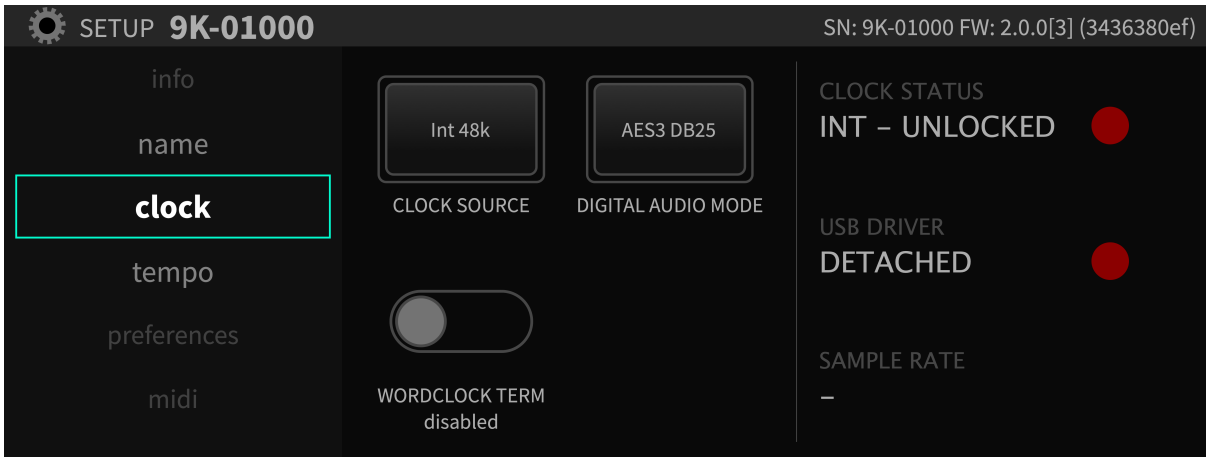


The *Name* screen allows you to create a custom name for your H9000. With custom names assigned, you can select the desired unit when using multiple units and controlling them with Emote.

To assign a custom name to your unit:

- Use the cursor keys to navigate to the *System Name* field.
- A “virtual number pad” is displayed, showing you which letters are available on each physical number key.
- Press the front panel number keys to enter the letters of the custom name you wish to give your unit.
- Press the number key repeatedly to cycle through each of the three letters.
- Press the arrow down key to toggle between uppercase and lowercase letters.
- Press the arrow up key to toggle between letters and special symbols.
- Press the *Enter* key to save the custom name. You will be asked to confirm your new custom name.
- Press the *OK* soft key to save the name, or *Cancel* to cancel the operation.

2.1.4. Clock



The *Clock* menu is where you will set the unit’s sample rate and digital audio mode.

Clock Source

The clock source is the digital audio clock to which your H9000 synchronizes. You can select an internal clock at one of the sample rates listed below:

- 44.1 kHz
- 88.2 kHz
- 48 kHz
- 96 kHz

Alternatively, you can synchronize your H9000's system clock to an external clock signal derived from additional sources:

- AES Any (any AES signal feeding the DB-25 connector)
- S/PDIF/XLR/AES in 1 (Dig 1-2)
- AES In 2 (Dig 3-4)
- AES In 3 (Dig 5/6)
- AES in 4 (Dig 7/8)
- ADAT
- Word Clock
- Expansion A/B/C

To adjust the clock source:

- Use the cursor keys to highlight the clock source setting.
- Press the *Enter* button or adjust the wheel to reveal up a popup menu with available choices for that setting.
- Use the cursor up/down buttons or wheel to select your desired choice.
- Press the enter key again to select that choice.

Digital Audio Mode

The H9000 contains several different digital audio connectors. They are "shared" as an input source to the H9000; you choose which one you wish to make active. The Digital Audio Mode menu is where this selection is made. Use the Digital Audio mode selector to pick your preferred connector, which will then be available in the audio routing menus of the H9000. Available choices include:

- AES 3 (DB-25 Connector)
- AES 3 (XLR Connector)
- S/PDIF (RCA Connector)
- S/PDIF (Optical Connector)
- ADAT (Optical Connector)

By contrast, for the outputs, the same signal is "multed" to all of the digital audio output connectors simultaneously.

Clock Status

The right-hand side of the screen displays the status of various clocks, as follows:

- Clock Status: The large virtual LED illuminates GREEN when the H9000 is locked to either its internal clock, or a valid external clock that you have selected for it to sync to. The LED illuminates RED if the H9000 is not locked. For example, if you have selected an external clock source that hasn't been connected.
- USB Driver: The large virtual LED illuminates GREEN when the H9000 has a valid USB connection to an attached computer and will display "attached." The LED illuminates RED if the H9000 has no valid USB connection and will display "detached."
- Sample Rate: This field displays the actual sample rate at which H9000 is operating.

> **Note**
>
> The H9000 does not support sample rate conversion. If you want to feed the H9000 a digital audio signal, the external digital signal and the H9000 must both be locked to the same digital clock, at the same sample rate.

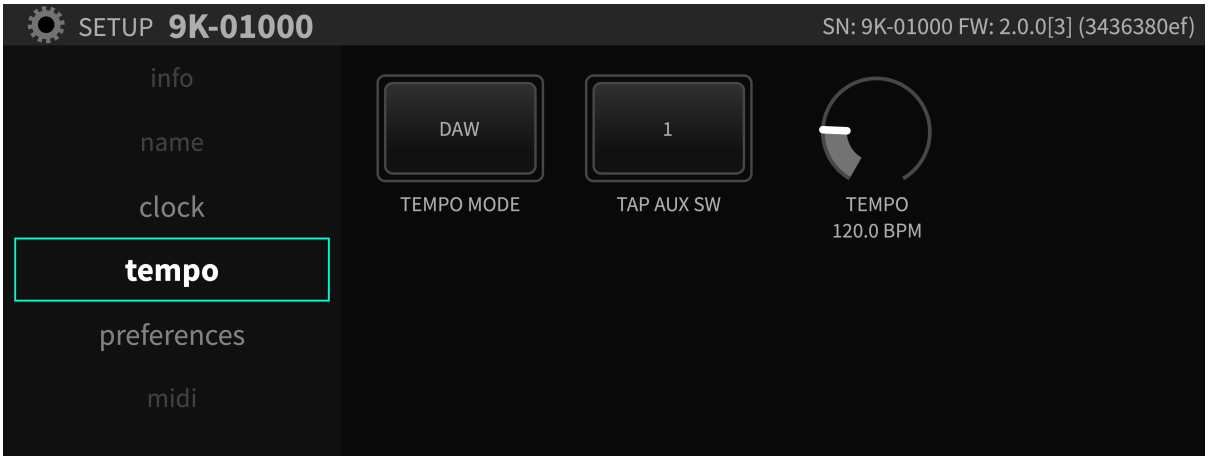
Word Clock Termination Status

Sets the internal word clock termination to be enabled or disabled.

- Set this to "enabled" if the H9000 is at the end of a word clock chain.
- Set this to "disabled" if the H9000 is in the middle of a word clock chain and a BNC T is being used.

> **Warning**
>
> Use a 75 ohm BNC cable. 50 ohm BNC cables may cause clocking issues.

2.1.5. Tempo



The *Tempo* menu provides controls for setting the H9000's tempo.

The unit's tempo affects the H9000's many time-based effects, which can be synchronized to various clock sources. For example, a delay algorithm's repeats can be synchronized to an incoming MIDI clock or your DAW. Tempo mode menu includes the following four settings:

Off

In this mode, no MIDI clock data is received from the USB or DIN inputs. Any time-based effects will not be synchronized to a MIDI clock signal.

Internal

In this mode, you set your own tempo, expressed as Beats per Minute (BPM). This BPM value is the tempo base for any relevant time-based effects. You can set the internal tempo in three different ways:

- Tap Tempo: Tap the front panel *Tap* button at least two times; the H9000 will automatically set the unit's internal tempo to the average of the most recent taps, and temporarily display the tempo in BPM on the display.
- TAP AUX SW: Highlight "TAP AUX SW" and press the *Enter* key to assign an aux switch to Tempo.
- Manual Tempo Dial: Highlight the Tempo setting, then adjust the wheel to manually select the BPM value you wish to set your tempo to.

MIDI Clock

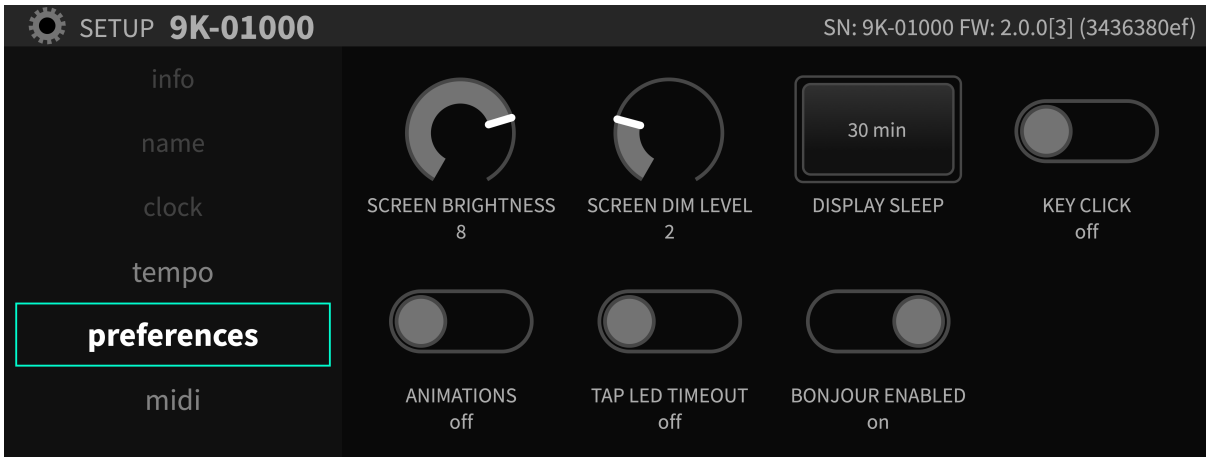
In this mode, the H9000 will automatically set the tempo to an incoming MIDI signal (as long as that MIDI signal contains a valid MIDI clock signal). The external MIDI clock signal can arrive either via DIN, or USB-B.

DAW

In this mode, the H9000 will automatically sync to your DAW's tempo. For this to work correctly, you must be using Emote as a plug-in in your DAW (see section 3.4).

> **Note**
>
> When tempo mode is set to Off, MIDI Clock, or DAW, the tempo dial and tap tempo functions will be disabled.

2.1.6. Preferences



The *Preferences* section controls various aspects of the H9000's display. Use the cursor keys to highlight the setting you wish to adjust, then use the *Enter* key and/or wheel to adjust the setting.

Screen Brightness

This control adjusts the brightness of the front panel screen on a scale of 1-10.

Screen Dim Level

This setting adjusts the brightness of the screen when it automatically dims after a selected amount of time, dubbed “Display Sleep”.

Display Sleep

This controls the amount of time after which the display will automatically dim.

Key Click

This setting toggles on/off an audible click when using the front panel buttons. The click is played from a small speaker located inside the H9000.

Animations

By default, the H9000 offers assorted on-screen animations for the front panel display. This control toggles the animations on/off.

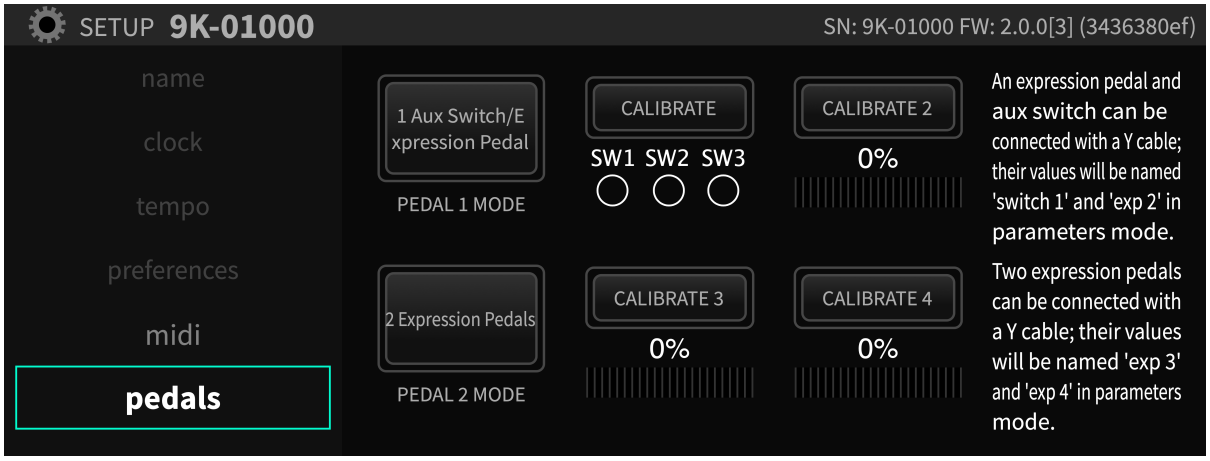
Tap LED Timeout

By default, the front panel *Tap* button flashes in time with the tempo you have set using the front panel Tap button. Change this setting to “On” if you wish the button to stop flashing a few moments after you have set the tempo.

Bonjour Enabled

Turning this setting to “On” configures the H9000 to advertise its services using the “Bonjour” protocol on the local area network. Bonjour is a zero-configuration network protocol that allows the H9000 to be automatically discovered by the Emote software, and should generally be left enabled for most network configurations.

2.1.7. Pedals



The H9000 allows connection of up to four independent foot pedals; these pedals can be a “switch” type (on/off) or a “continuous” type (volume) pedal.

When setting up an FX chain and its underlying algorithms, you can map the action of the footswitch/foot pedals to any parameter of your choosing, allowing expressive control of the effects. For example, a guitar player can control the cutoff frequency of a resonant filter using a foot pedal, while playing their instrument.

The *Pedals* setup page allows you to configure your unit for a wide variety of different types of pedals available in the marketplace, taking into consideration factors such as:

- On/Off Switch vs. Continuous pedal
- How many switches are on the foot pedal, which can range from none to as many as three switches per pedal
- “Polarity” of the attached pedal

2.1.7.1. Configuring your Attached Pedal

- Use the cursor buttons to highlight the “Pedal 1 Mode” or “Pedal 2 Mode” mode.
- Press the *Enter* key, then use the cursor keys or wheel to navigate among the pop-up menu of options. Available choices include:

  - Expression Pedal
  - 2 Expression Pedals
  - 1 Expression Pedal + 1 Aux Switch
  - 1 Aux switch + 1 Expression Pedal
  - 2 button Aux Switch
  - 3 button Aux Switch

After you have selected the mode appropriate for your connected switch/pedal, press the *Enter* key to save your selection.

2.1.7.2. Calibrating Your Attached Pedals and Switches

Once you have selected your desired type of pedal/switch, the H9000 will populate the rest of the display with relevant calibration tools.

- Use the cursor button and the Enter key to navigate to and engage the “Calibrate” field.
- If you have attached an on/off type pedal, physically press each switch and make sure that you see the appropriate “SW” circle illuminate. There are separate SW circles for 1, 2, or 3 button pedals. 3 SW circles are shown for 1, 2, or 3 button pedals because even a single aux switch may not be sent out as SW1 but may instead be sent out as SW2.
- If you have attached a continuous foot pedal, navigate to the “Calibrate Pedal” field, then sweep the pedal back and forth with your foot. You should see the Calibrate Pedal field display a range between 0-100% on the calibration meter.
- When you’ve finished calibrating, disengage the “Calibrate” button using the Enter key.

Even though there are only two pedal/switch connectors on the H9000, you can connect up to four pedals. This is made possible by connecting two pedals to each connector using a “Y” cable: https://etide.io/TRS

This capability is possible because:

- In the external control section of the Parameters mode, there are 4 expression pedals listed.
- In set-up mode, there are options to configure a single connector to be connected to two separate expression pedals.

2.1.7.3. Control Voltages

The H9000 can also be controlled using up to 4 control voltage sources. To do so, follow the instructions for setting up multiple expression pedals using a “Y” cable. CV 1+2 will correspond to the TIP and RING of Expression Pedal 1, CV 3+4 will correspond to the TIP and RING of Expression Pedal 2.

2.1.8. MIDI

The *MIDI* menu displays options to enable and disable MIDI feedback.

FEEDBACK DIN

Sends MIDI from the MIDI-in DIN jack directly to the MIDI-out DIN jack.

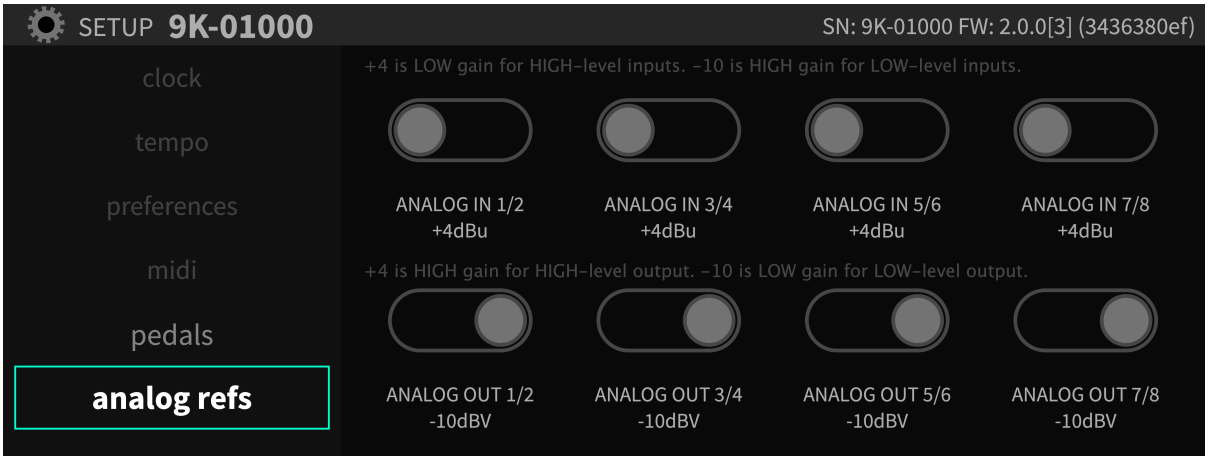
FEEDBACK USB A

Sends MIDI from MIDI controllers plugged into the USB-A ports back to those very same MIDI controllers.

FEEDBACK USB B

Sends MIDI from a USB host plugged into the USB-B port back to the very same USB host.

2.1.9. Analog References



The Analog Reference Levels screen allows you to set the levels for the eight analog inputs and outputs present on the DB-25 connectors. This allows the H9000 to be installed in a wide variety of environments, and interface at an optimum level with professional and some consumer equipment.

Each pair of analog inputs and outputs on the DB-25 connectors can be separately selected between -10 dBV and +4 dBu.

Use the cursor buttons to highlight the specific pair of inputs or outputs you wish to adjust. Options include:

- Analog In 1/2
- Analog In 3/4
- Analog In 5/6
- Analog In 7/8
- Analog Out 1/2
- Analog Out 3/4
- Analog Out 5/6
- Analog Out 7/8

Once you have highlighted the appropriate pair of inputs or outputs, press the *Enter* button to switch to the alternate operating level.

Note that the rear panel XLR analog connectors are a duplicate “mult” of channels 1-2 of the eight-channel analog DB-25 connectors. As a result, any reference level changes made to analog input or output 1-2 of the DB-25 connector will also affect the XLR inputs and outputs.

2.1.10. Network

The network screen controls settings for connecting your H9000 to a standard local area network. Once a network connection is established, you can use the Emote software to view and control H9000 parameters and update your H9000's firmware.

The middle column of the network screen shows the available network connections, including:

- Ethernet (for a hard-wired connection directly to the computer, or through your local area network)
- Wireless (for a connection using the included Wi-Fi adapter, which can be connected to any of the four USB connectors)

2.1.10.1. Ethernet Setup

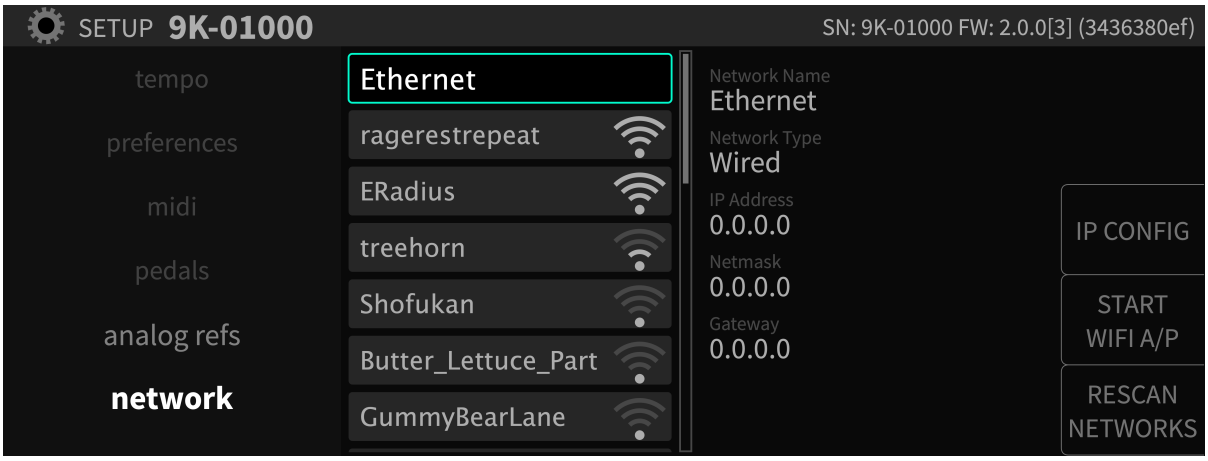
Use the cursor buttons to highlight the Ethernet field in the middle column. The right-hand column will display the parameters of the network you have joined including:

- Network Name
- Network Type (wired or wireless)
- IP Address
- Netmask
- Gateway

2.1.10.2. Wireless Setup

To connect the H9000 to a wireless network:

- Connect the included Wi-Fi adapter to one of the H9000's four USB ports.
- On the network screen, any available wireless networks will be automatically listed beneath the *Ethernet* option.



- Navigate to the wireless network that you wish to connect to and press the *Enter* key.
- If the network is not password protected, you will be connected to the wireless network.
- If the network is password protected, you will be prompted to enter the appropriate password. Use the front panel numbers pad to enter the password, then press the *OK* soft key.

- You will then be connected to the wireless network, and can use the Emote software to control the H9000.

Note that the H9000 stores the authentication settings for the most recently joined network, so if you lose connection for any reason, you can re-join the network without having to enter your password again.

2.1.10.3. Network Configuration

To change your H9000's network configuration, press the *IP CONFIG* soft key. Choose "DHCP" to initiate a request for an IP address, or "Manual" to configure a static address.

Note

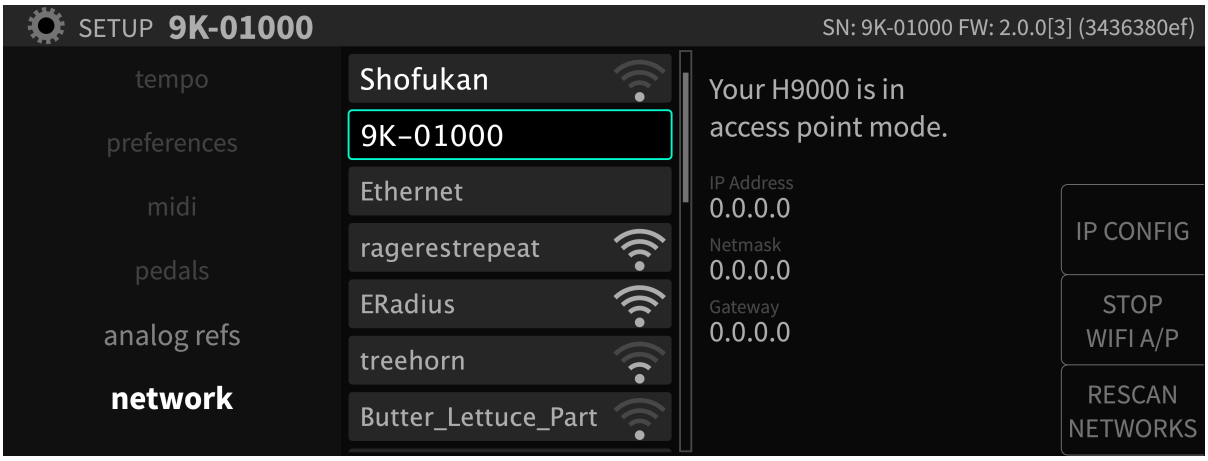
Your H9000's network configuration settings will be set to DHCP as the default setting and will not require further configuration in most situations. Manual IP addressing should only be used if there is no DHCP server on the network, or the DHCP server is configured to exclude a range of manual addresses.

2.1.10.4. Creating a Local Access Point

The H9000 also has the ability to create and broadcast its own Wi-Fi network, called an "access point". You can use the Emote software with the H9000 on this private network in environments where there is no LAN for the H9000 and the computer running Emote to connect to each other.

To create an Access Point:

- Press the *START WIFI A/P* soft key.
- You can choose to enter a custom password, or leave the password field blank.
- If you leave the password field blank, a password will automatically be generated. This will be your H9000's serial number in the format "9K-XXXXX".
- You will then see a message that the Access Point has been created.
- The Wi-Fi access point will appear in your computer's network settings as "H9000" followed by a string of automatically generated characters, or the custom name you have assigned to your unit.



Warning

Turn off the access point after you are finished using it. Otherwise it may interfere with Emote's connectivity.

2.2. Using the H9000 as a Computer Audio Interface

The H9000 can stream 16 inputs and 16 outputs via USB 2.0 to/from your Windows or macOS DAW, serving as a powerful outboard effects processor for your DAW's tracks.

2.2.1. Changing the USB Audio Sample Rate

When using the H9000 as a USB audio device, the H9000 is always the USB clock source and determines the sample rate. Since this information is not automatically communicated to the computer, the user must ensure that the H9000's internal sample rate matches the "H9000 USB Audio" sample rate in your computer's audio settings. It is not possible for the H9000 USB Audio sample rate to automatically follow the H9000's internal clock or an external clock source (Word Clock, ADAT, etc.), and the user must always change the H9000 USB Audio sample rate manually.

When switching to a new sample rate, first change the H9000's sample rate (by selecting a different internal clock source or changing the sample rate of an external source such as ADAT, Dante, etc.), then change the H9000 USB audio sample rate in your computer's audio settings.

If the H9000 detects that the sample rates don't match, it will mute the USB audio interface and display a warning on the front panel, and in Emote. When the sample rates match again, the warning message will be cleared and the audio un-muted.

If the H9000's internal sample rate matches the H9000 USB Audio sample rate but the USB Audio Mute warning is still displayed, then toggle your computer's H9000 USB Audio sample rate away from, and then back to, the H9000's sample rate. This will resync the sample rate of the computer and H9000.

2.2.2. Using the H9000 with macOS

The H9000 works as a "plug and play" Core Audio device in macOS, versions 10.11 and higher. Connect a USB cable between the USB B connector on the H9000 and an available USB A port on your macOS computer. No driver installation required!

The H9000 will now be available to all CoreAudio compatible software as a 16-input, 16-output computer audio device, and can be operated at 44.1kHz, 48kHz, 88.2kHz, and 96kHz.

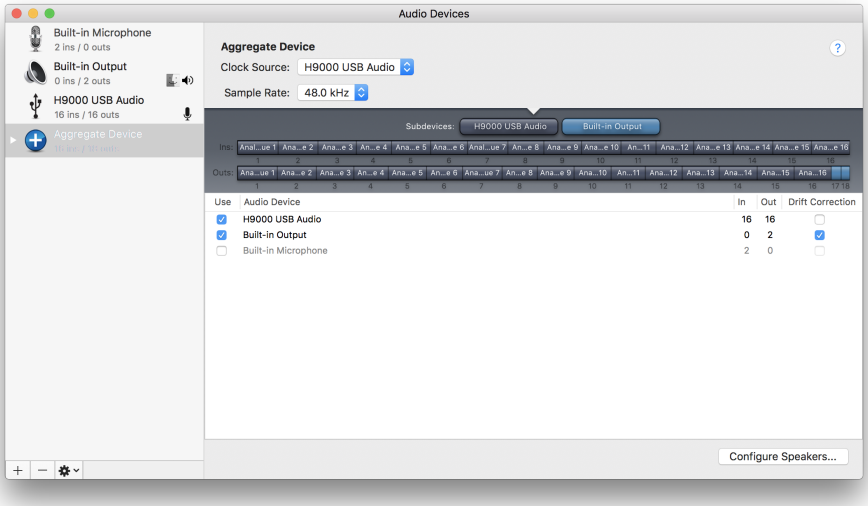
2.2.2.1. Using the H9000 With Your Existing Computer Audio Interface

The H9000 is an outboard effects processor, not a traditional, standalone computer audio interface with control room monitoring functions. As a result, you will likely want to use the H9000's computer connectivity in conjunction with your main computer audio interface.

Fortunately, the macOS Core Audio system makes it easy to combine multiple audio interfaces into one large device called an "Aggregate Device." Doing so lets you use your main audio interface and its monitoring capabilities as you always have, while also having the H9000's 16 channels of I/O available to process tracks on your DAW session.

To Create an Aggregate Device:

- In your applications folder, go to "Utilities," then launch "Audio Midi Setup."
- In the "Audio Devices" window that appears, you should see entries for:
  - H9000 USB Audio
  - Any other computer audio interfaces you have connected
  - The computer's built in audio output



- Click on the “+” Icon in the lower left-hand corner and select “Create Aggregate Device”.
- In the Aggregate Device configuration window that appears, select the appropriate I/O for the different devices you wish to use.
- In your DAW of choice, select the Aggregate Device as your audio device for the DAW to use.
- You will now have the I/O of your main audio interface AND the 16-input/output channels of the H9000 available to use in your DAW session as needed.
- Ensure that the sample rates match on the following:
  - The H9000’s internal clock
  - The Aggregate Device
  - The H9000 USB Audio Device
  - Your Digital Audio Workstation
- If the devices in your Aggregate Device aren’t synchronised using hardware, you must enable drift correction to compensate for drift in the data between devices. Please visit https://etide.io/mac-midi for more information about configuring drift correction.

⚠ Warning

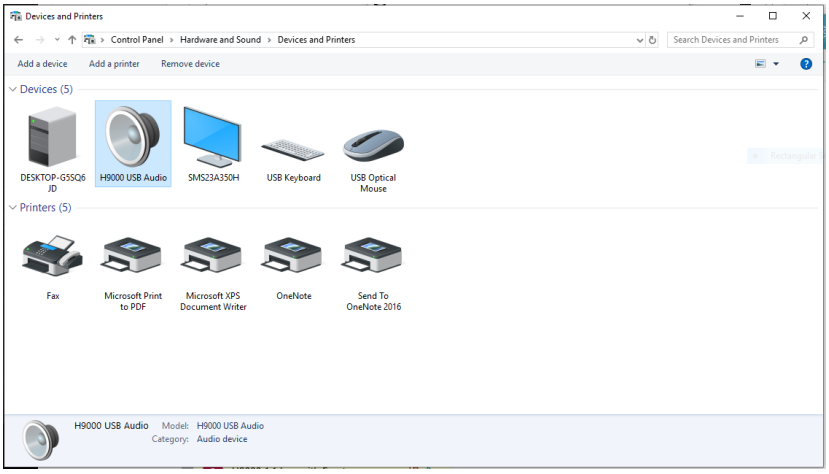
Aggregate devices don’t always perform as well as when the interfaces are alone. You might experience digital clock source problems, dropouts, noise, or other playback related issues. If you experience poor performance or connectivity issues, an aggregate device may not be ideal.

2.2.3. Using the H9000 with Windows

The H9000 is compatible with computers running Windows 10 or 11.

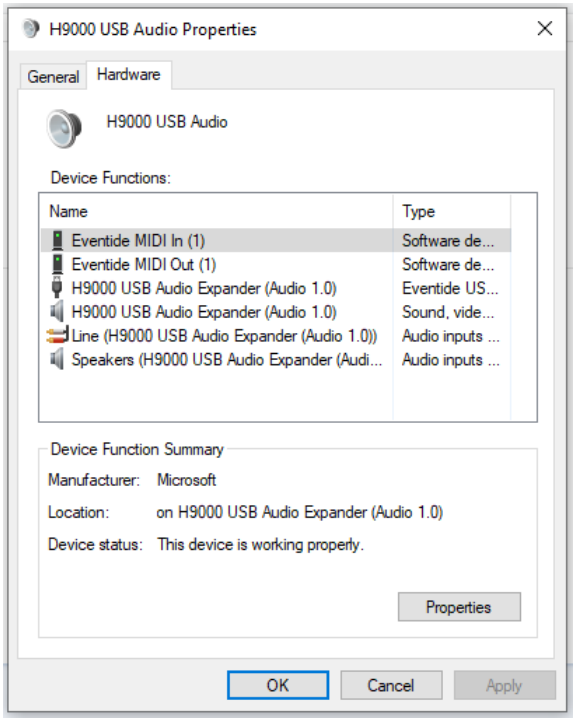
In order to use the H9000 as an audio interface with Windows, please visit https://etide.io/h9000-support to download and install the Windows driver.

After downloading the .exe file, double-click the file to install it on your Windows machine and reboot your computer.



The H9000 will now appear as an audio device in the Windows “Devices and Printers” control panel, and its audio inputs and outputs can be used with any ASIO compatible software.

The H9000’s MIDI inputs and outputs are also now available to any software that works with MIDI ports.



Depending on the setup of your Windows machine, the H9000's USB sample rate can be changed one of two ways.

1. Windows OS method:

   Use this method when using the H9000 with applications that don't use ASIO drivers. For example, if you're using the H9000 USB audio as your computer's primary audio interface and you are not using a DAW.

   - Click the Windows menu bar icon on the bottom left of your screen.
   - Type "Sound Settings" and click the search result. The Sound Settings Window will appear.
   - In the Sound Settings Window, under "Related Settings", click "Sound Control Panel." The Sound Control Panel will appear.
   - In the Sound Control Panel, under the "Playback" tab, right-click "H9000 USB Audio Expander" and click "Properties". The Speakers Properties Window will appear.
   - In the Speakers Properties Window, click the "Advanced" tab. Select the appropriate sample rate, click "Apply", then click "OK".
   - Back in the Sound Control Panel, under the "Recording" tab, right-click "H9000 USB Audio Expander" and click "Properties". The Line Properties Window will appear.
   - In the Line Properties Window, click the "Advanced" tab. Select the appropriate sample rate, click "Apply", then click "OK".

2. DAW method:

   Use this method when you are using the H9000 with ASIO drivers in a DAW.

   - Set your DAW's Audio Device to "Eventide USB Audio Device" and select the appropriate sample rate.
   - You may need to toggle this setting to another sample rate, and then back to the desired sample rate to resync the USB audio device.

This page was intentionally left 99.88% blank.

3. EMOTE

Emote is a stand-alone program for:

- macOS computers running 10.11 or higher (Apple Silicon (M1/M2) native is supported).
- PCs running Windows 10 or higher.
- VST3, VST, AU, and AAX plugin formats.

Emote gives you control of all of the features and settings of your H9000 effects processor, allowing you to craft your effects quickly and intuitively. Think of Emote as your artistic partner, letting you dig deeper into the creative potential of the H9000.

3.1. Getting Started

3.1.1. Downloading the Installer

Please visit https://etide.io/emote to download the production release of Emote. Once the installer is downloaded, double click the installer icon and follow the on-screen instructions. The installer allows you to choose which formats of Emote to install, as well as the install paths.

- On macOS, since Emote is manually downloaded from our website (as opposed to the macOS app store), you will be prompted to approve the installation of the program during the install process.
- On Windows, Emote is digitally signed by Microsoft, so you shouldn't see any extra security prompts while installing.

Note

On Windows, visit https://etide.io/DL999 to download Apple Bonjour Print Services, then restart your PC. This installs a system component for finding any Bonjour services on the network, which allows your H9000 to show up in Emote automatically without needing to manually enter an IP address.

3.1.2. Connection

Emote can connect to your H9000 in three ways:

1. Ethernet, directly to your computer.
2. Ethernet, through your local area network.
3. Wirelessly, by using the included Wi-Fi adapter.

> **Note**
>
> Emote cannot connect to the H9000 via USB, but it does display the settings for, and the current state of, the H9000's USB Audio and MIDI.

3.1.2.1. Connecting to Emote With an Ethernet Connection

Plug one end of an ethernet cable into the H9000, then plug the other end of the ethernet cable into either your computer or your router.

Power on your H9000, and open Emote. The H9000 will appear in Emote in a couple minutes, once it fully boots.

After connecting to Emote with an ethernet connection, you can continue to use ethernet, or alternatively you may select a Wi-Fi network for use with the included Wi-Fi adapter.

3.1.2.2. Connecting the H9000R to Emote With a Wi-Fi Connection

If no ethernet is present, you must create an initial connection between the H9000R and Emote by using a Wi-Fi access point.

1. Plug the included Wi-Fi adapter into any USB port on the H9000R.
2. Turn on the H9000R, while simultaneously pressing the front panel "Network" button. This will activate the H9000R's Wi-Fi access point at startup. Note that booting with the "Network" button held down will also implement the Clear Current State (Section A.1) recovery function. This does not affect any of your saved sessions or presets, but will clear the currently loaded session and all settings in Setup View, including your network and authentication settings.

3. The network button's LED will flash for about two minutes, then be dim for a few seconds, then brighten. At this time a wireless access point will have been created.
4. The Wi-Fi access point will appear in your computer's network settings as "H9000" followed by a string of automatically generated characters, or the custom name you have assigned to your unit.
5. You will be required to enter an automatically generated password to connect to the Wi-Fi Access Point. This will be your H9000R's serial number in the format "9K-XXXXX".
6. Connect to this ad-hoc network on the computer that is running Emote. At this point the H9000R should appear as a device in Emote's browser window. Click on the device in Emote's browser to connect.

Warning

Turn off the access point after you are finished using it. Otherwise it may interfere with Emote's connectivity.

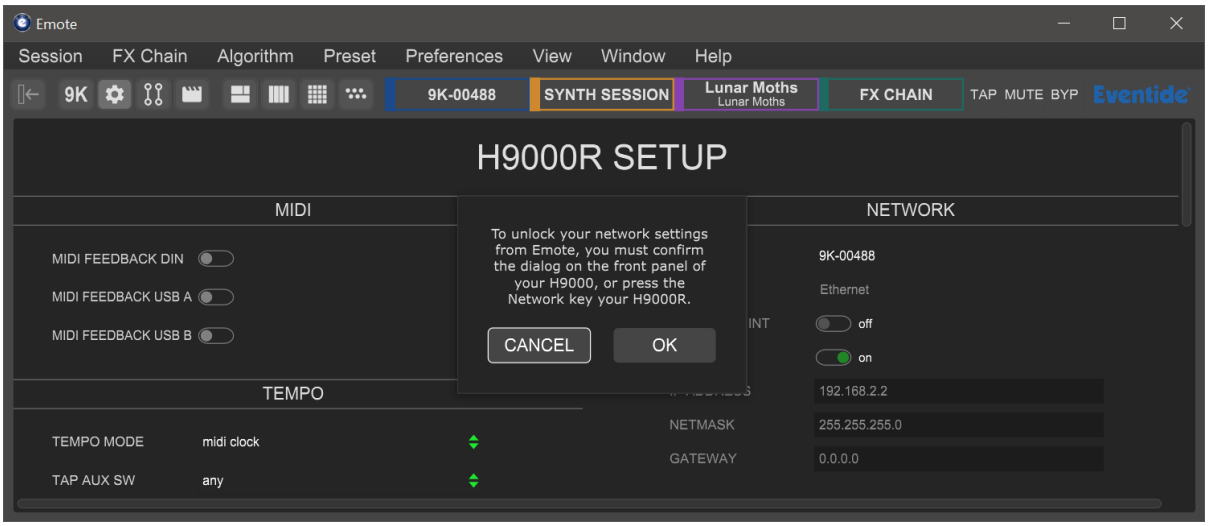
3.1.2.3. Switching Networks Using Emote

Note

To avoid connection issues, restart Emote whenever you change your network settings from ethernet to Wi-Fi or vice versa.

Switching networks using Emote requires a network handshake between the Emote software client and the physical H9000 device. This is done in order to prevent accidental disconnections, as well as any potential malicious H9000 connections.

The Network field in Setup View will be greyed out. Click the field to bring up on-screen instructions on how to unlock it. Click "OK" in the resulting dialog box.



You will then have 30 seconds to confirm the dialog on the front panel of your H9000, or press the Network button on your H9000R. The network selection field in Emote will be unlocked, and it will now no longer be greyed out. This generates a temporary encryption key, which is used to protect usernames and passwords sent via Emote. You will then have 2 minutes to select and connect to your main network, after which you would need to re-do the handshake process to unlock the network selection field again.

When the network connects, quit the Emote software. Re-connect your computer to your main network, and re-open Emote. The H9000 should now be visible in Emote's browser. If it isn't, then reboot your H9000.

⚠ Warning

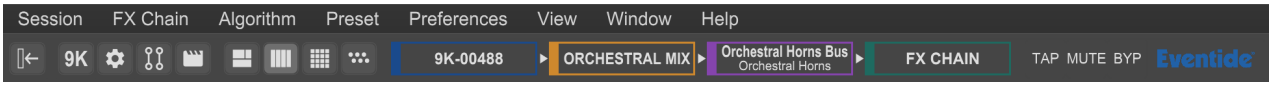
Turn off the access point after you are finished using it. Otherwise it may interfere with Emote's connectivity.

3.2. Overview

Emote's interface contains three key areas:

- Menu bar
- Tool Bar
- Browser

3.2.1. Menu Bar



The Menu Bar offers the following menus:

1. Session

- Open: `ctrl/⌘` + `o` Open a session from a file you have saved to your computer.
- Open Recent: Open a Session that you have recently opened from a file you have saved to your computer.
- Save: `ctrl/⌘` + `s` Save the currently open Session and keep the same name and location.
- Save As: `ctrl/⌘` + `shift` + `s` Save a Session file to your computer.
- Save To H9000: `ctrl/⌘` + `alt` + `s` Save a Session file to the connected H9000.

2. FX Chain

- Open: Open an FX Chain from a file you have saved to your computer.
- Save: Save the currently selected FX Chain and keep the same name and location.
- Save As: Save an FX Chain file to your computer.
- Save To H9000: Save an FX Chain file to the connected H9000.

3. Algorithm

- Import Algorithm: Currently only supports the .9ka file format. To upload algorithms in the .sig2 file format, use Vsig3 instead of Emote.
- Export Algorithm: Exports the currently selected algorithm as a .9ka file. This is useful if you have created a custom algorithm and would like to export it to share with other users.

4. Preset

- Open: Open a Preset from a file you have saved to your computer.
- Save: Save the currently selected Preset and keep the same name and location.
- Save As: Save a Preset file to your computer.
- Save To H9000: Save a Preset file to the connected H9000.

5. Preferences

- Reset Session Load Options

- Reset All Dialogs

6. View

- Toggle Browser `ctrl/⌘` + `[`
- Zoom In `ctrl/⌘` + `=`
- Zoom Out `ctrl/⌘` + `-`
- Reset Zoom `ctrl/⌘` + `0`

- Show:

  - Devices View `5`
  - Setup View `6`
  - Direct I/O View `7`
  - Scenes View `8`
  - Combined View `1`
  - FX Chains View `2`
  - Algorithms View `3`
  - Parameters View `4`

- Select:

  - FX Chain 1 `ctrl/⌘` + `1`
  - FX Chain 2 `ctrl/⌘` + `2`
  - FX Chain 3 `ctrl/⌘` + `3`
  - FX Chain 4 `ctrl/⌘` + `4`
  - Algorithm 1 `alt` + `1`
  - Algorithm 2 `alt` + `2`
  - Algorithm 3 `alt` + `3`
  - Algorithm 4 `alt` + `4`
  - I/O `alt` + `` ` ``
  - Session Tab `Q`
  - Chain Tab `W`
  - Algo Tab `E`
  - Preset Tab `R`

7. Window

- Minimize: `ctrl/⌘` + `M`
- Meter Bridge: Displays all of the H9000's hardware I/O meters in a separate window that allows you to view any physical I/O source or destination in banks of 8. This is useful for debugging setup issues or doing more precise gain staging, as these meters are hardware-based, unlike the software metering provided within each FX chain.

8. Help

- User Guide: `ctrl/⌘` + `G`
- About Emote: Shows Emote's version.

3.2.2. Tool Bar

The tool bar allows you to select between Emote's Views and offers the following options (from left to right):

1. View Options

- Show/Hide Browser Panel `ctrl/⌘` + `[`
- Show Devices View `5`
- Show Setup View `6`
- Show Direct I/O View `7`
- Show Scenes View `8`
- Show Combined View `1`
- Show FX Chains View `2`
- Show Algorithms View `3`
- Show Parameters View `4`

2. Color Blocks

- Show Devices View `5`
- Show Combined View `1`
- Show FX Chains View `2`
- Show Parameters View `4`

3. Globals

- TAP: Click the TAP button a single time to temporarily display the current tempo used by the H9000. Click the TAP tab repeatedly to manually set a new tempo.
- MUTE: Click the MUTE tab to toggle global mute on/off. This does not affect Direct I/O.
- BYP: Click the BYP tab to toggle global bypass on/off.
- Eventide: Hover your mouse over the Eventide logo to view Emote's version number.

Note

Keyboard shortcuts are only supported in the standalone version of Emote.

3.2.3. Browser Window

The Browser Window may be shown/hidden in Devices View, Combined View, FX Chain View, Algorithm View, and Parameter View.

To load a Session, FX Chain, Algorithm, or Preset, drag it from the Browser Window and drop it onto the appropriate spot in the middle of the main screen. This dragging concept also extends to other areas such as moving algorithms, or creating Input/Output connections. Emote makes use of the "drag and drop" method for most operations. Alternatively, you can double-click any item in the Browser Window to load it, or select it with your mouse or arrow keys and press `RETURN`

3.2.3.1. Adding and Removing Sessions

- Use the mouse to drag and drop a session (preset or user-created) from the browser to the middle of the screen. The session will then load.
- If you have a session file on your computer that you'd like to open in Emote, you can go to Emote's Session menu and click Open. From there, you can browse your computer's file system to find your desired session. These files will be prefixed with `.9ks`
- You can also type text into the search box at the top of the browser to find a particular session. Click the "X" to clear the search box and show all sessions.
- To delete a session permanently, right-click the session in the browser and click delete.
- To clear a session, either load one of the default sessions numbered 1001-1008, or delete everything in the current session. Note that at least 1 FX Chain must always be present within a Session.

3.2.3.2. Adding and Removing FX Chains

- Use the mouse to drag and drop an FX chain (preset or user-created) from the browser to the specific FX chain slot you wish to populate. The FX chain will then load.
- You can also type text into the search box at the top of the browser window to locate a particular FX chain. Click the `X` next to the search box to clear the search field and show all FX chains.
- To remove an FX chain from a session, drag the FX chain you want to remove back to the browser window on the left and drop it. The FX chain will then be removed, after first prompting you with an "Are You Sure?" message.

- To delete an FX chain permanently, right-click the FX chain in the browser and click delete.
- To clear an FX chain, either load one of the default template FX chains numbered 1601-1615, or delete all algorithms and audio connections in the FX chain. Note that at least 1 algorithm must always been present within an FX Chain.

3.2.3.3. Adding and Removing Algorithms

Type text into the search box at the top of the Browser window to locate a particular algorithm. Algorithms that match your search will appear in the browser. If the algorithms have presets then the number of presets will be shown in a circle to the right of the algorithm's name. Click the circle to show and hide the algorithm's presets. Click the `X` next to the search box to clear the search field.

Emote also offers powerful tag filtering capabilities, built to quickly narrow the choices from the large number of available algorithms.

- In the algorithm tab of the browser window, you will see a list of available tag categories to choose from. Categories include Effect Type, Product Type, Instrument Type, and Inputs/Outputs.
- Click on a category to view its contained tags, then click on a tag to toggle it on/off.
- The browser will now display algorithms that match all of the tags that are currently highlighted in red.
- To remove tags, click on your previously selected tags to un-highlight these selections.
- Clicking "Sort By" allows you to sort algorithms by ID ascending, ID descending, name, most used, and last used.
- Right-click an algorithm in the browser and select "more info" to display its description.

3.3. Views

Emote's interface provides various Views that focus on different aspects of the H9000's controls.

3.3.1. Devices View

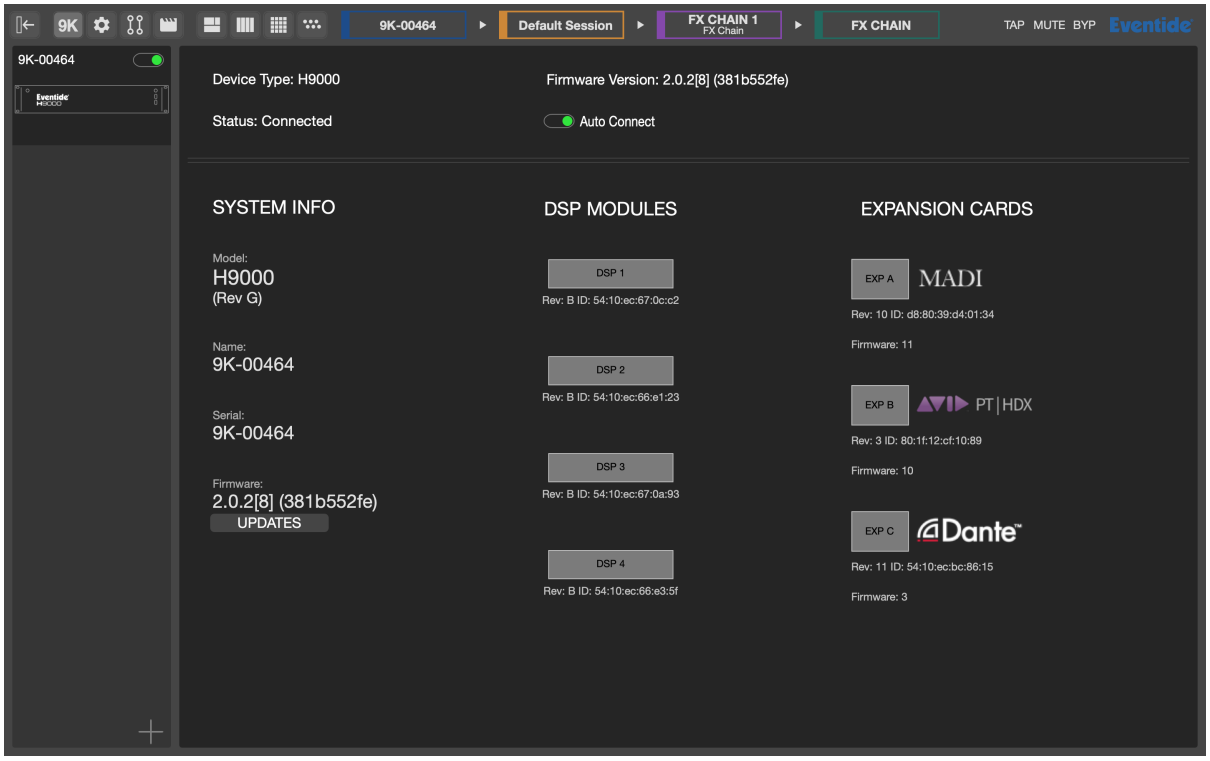


Fig. 3.1 **Devices View**

In Devices View, the Browser window displays the available H9000s present on the local area network. Click the “Connect” toggle above the H9000 icon to connect to the desired H9000. Emote allows you to connect to multiple H9000 units at once. You can then click in the Browser on the specific unit that you wish to control at any given time. You can also activate the Auto-Connect feature of a given H9000 to have it automatically connect to that unit when Emote is launched on a specific machine.

- Click on the “+” Icon at the bottom of the Browser to manually enter the IP address of an H9000 on the network. This feature can be useful if Emote is not displaying a unit that you know is available on the network.
- Right-click the H9000 device icon to download a system report. This may be useful when requesting customer support.
- Devices View displays information for the currently selected H9000, such as firmware version, number of installed DSP modules, types of expansion cards and their firmware versions, etc.

System info

- Model Number: Displays the H9000's model number and Rev version.
- Name: Displays a custom name that can be assign to the H9000.
- Serial: Displays the H9000's serial number.
- Firmware: Displays the H9000's currently installed firmware version.
  - UPDATES: Update your H9000's firmware.
  - DUMP DATA: Export all saved Sessions, FX Chains, Algorithms, and Presets from your H9000 to a connected FAT32-formatted USB drive before updating.

> **Note**
>
> Please update the H9000's firmware before updating the Emote software. Otherwise, Emote may not be able to connect to the H9000.

DSP modules

Displays the H9000's four DSP Modules. If a DSP module is displayed as (unoccupied) or OFFLINE, please refer to Section A.5: USB Self-Test Mode.

Expansion Cards

Displays any installed expansion cards and information about them, e.g., firmware version.

3.3.2. FX Chains View

FX Chains View allows loading of specific FX chains into available slots and also offers adjustments of FX chain settings.

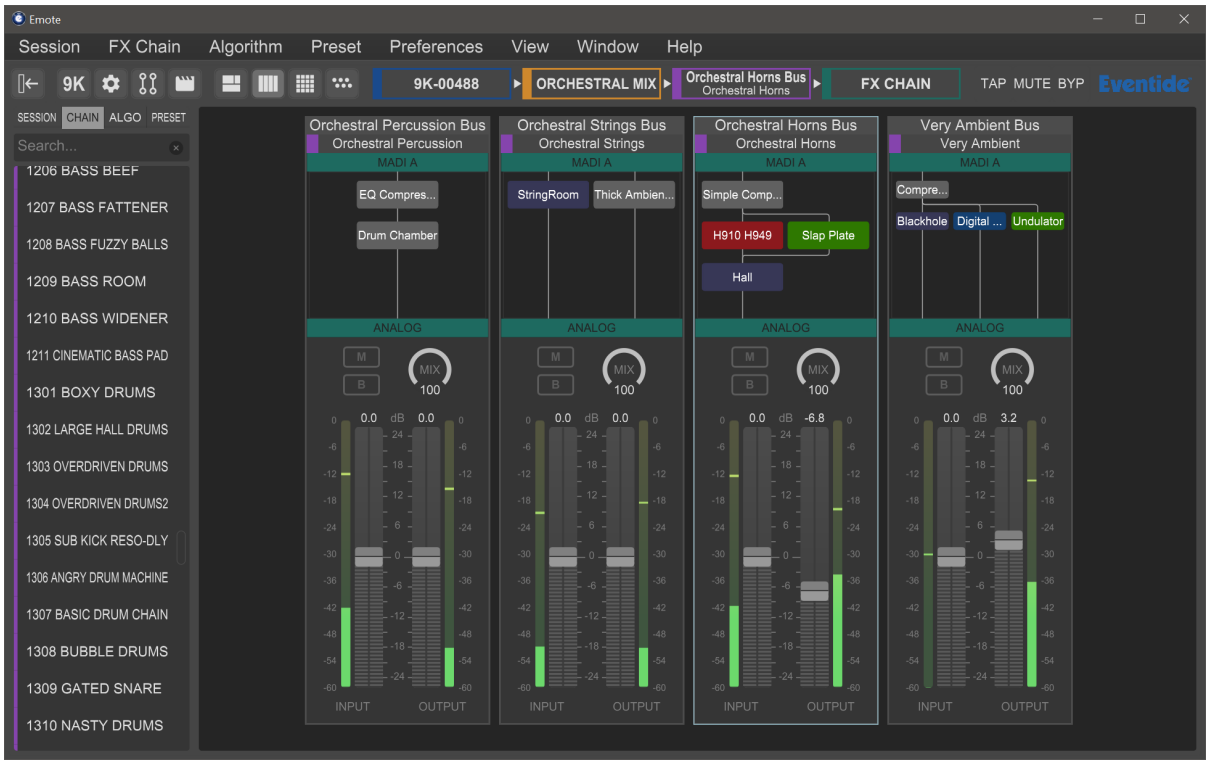


Fig. 3.2 **FX Chains View**

3.3.2.1. Adjusting FX Chain Settings

- Use the mouse to adjust the on-screen controls for each FX chain, including:
  - Mute
  - Bypass
  - Input and output levels
  - Mix
- `Alt/Option` + click the level faders or mix control to return them to their default state.
- Double-click the FX chain name to type a new title, then press the `RETURN` key to save the new name.
- Double-click the FX chain container name to type a new title, then press the `RETURN` key to save the new name.
- Click on an individual algorithm shown in the FX chain to adjust its parameters.
- Click on the input or output block of an FX chain to adjust the parameters for the audio input/output faders, and the functions for that FX chain.

3.3.3. Algorithms View

Algorithms View allows you to:

- Edit input/output routings to/from an FX chain.
- Edit algorithm routings within an FX chain.
- Mute/Bypass/Solo individual algorithms.

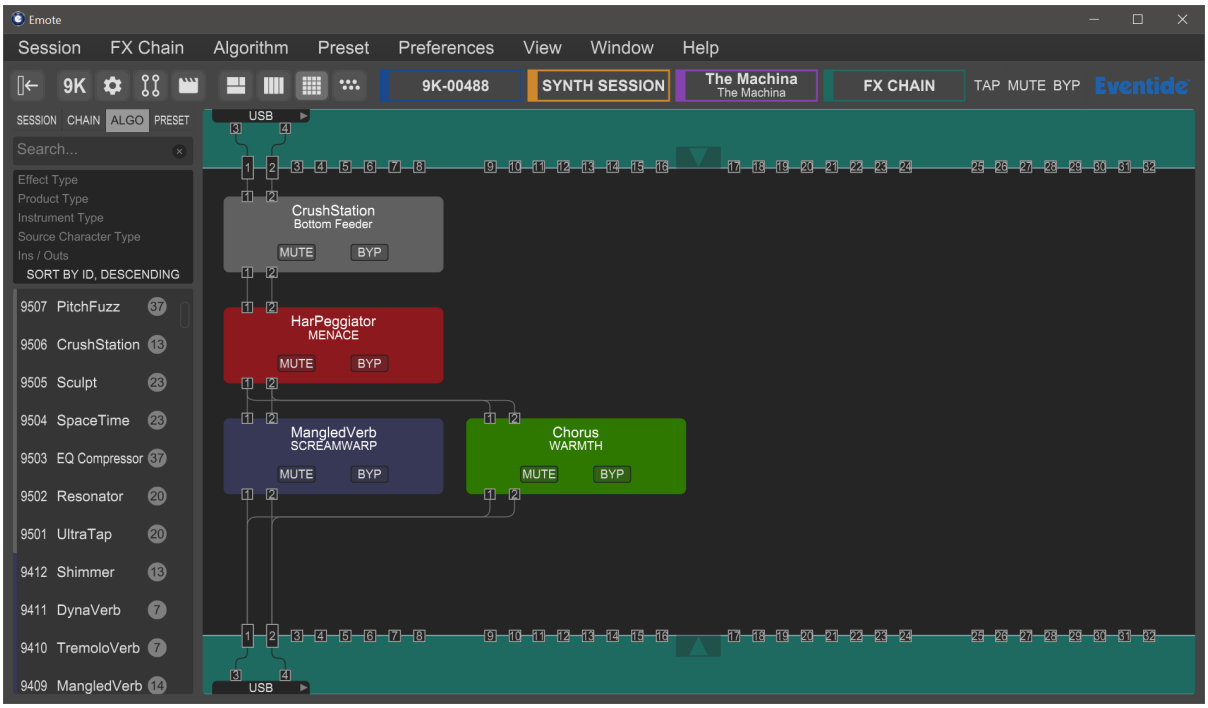


Fig. 3.3 **Algorithms View**

3.3.3.1. Editing Algorithm Routing Within an FX Chain

- Click and drag an algorithm to a blank area of the main screen to move or copy that algorithm to one of the other available slots.
- Click and drag an algorithm to another algorithm slot to “swap” the position of the two algorithms.
- Right-click an algorithm to show a menu that gives you the options to delete the algorithm from the FX chain, locate the algorithm within the browser window, or remove all I/O connections from the algorithm.

Note

FX Chain 1 automatically routes newly added algorithms. FX Chain 2, 3, and 4 do not do this.

3.3.3.2. Configuring Audio Connections

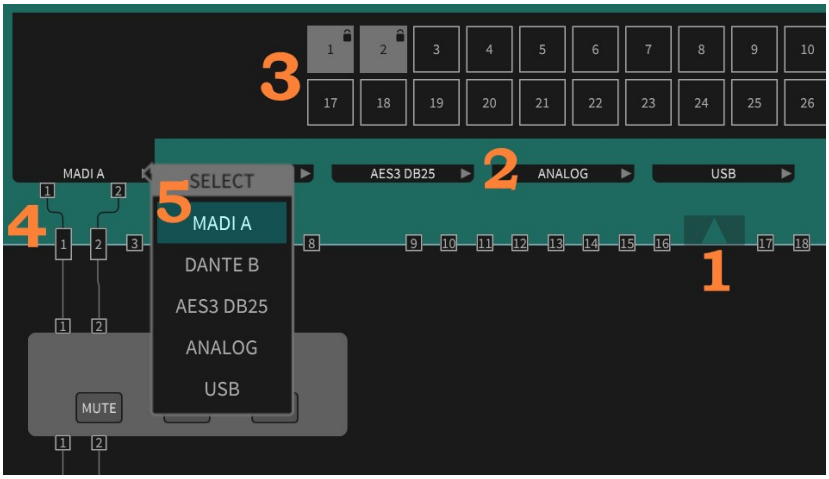


Fig. 3.4 **FX Chain I/O**

1. Click on one of the triangles at the top or bottom of the screen to show the audio I/O channels.
2. Click on an audio I/O group (USB, AES/EBU, Analog, etc.) to show its available channels.
3. Activate a channel either by clicking it, or by holding `alt/option` and moving the mouse over it. If the channel is locked, `shift`-clicking will unlock it. The active channel will show up as a routing block bordering its group.
4. Click, or click-drag, to make connections between the group routing blocks and the FX chain routing blocks. Once a connection is made, the channel will display a lock icon and be greyed out. Clicking blank space in the FX chain, or hitting the `Esc` key, ends the connection-making process. Right-click a routing block to remove that block's connections.

> **Note**
>
> Between FX chain routing blocks and Algorithm routing blocks, connections can be multed to multiple destinations, summed from multiple sources, and made in series or parallel.
>
> Between I/O group routing blocks and FX chain routing blocks, any input can be multed to multiple outputs, but multiple inputs cannot be summed to one output.

5. I/O channel activations can be transferred from one group to another by right-clicking an I/O group. A menu of eligible I/O groups to transfer to will appear. This feature is not yet supported in Direct I/O View.

3.3.4. Parameters View

Parameters View allows you to adjust specific parameters within an FX chain or algorithm, and assign MIDI, expression, and Function mappings.

- Hold `Alt/Option` while clicking on a parameter to reset it to its default value.
- Hold `Alt/Option`, `Shift`, and/or `ctrl/⌘` while scrolling over a parameter to adjust it.
- Hold `Shift` while adjusting a parameter for fine control.
- Hold `ctrl/⌘` while adjusting a parameter for ultra-fine control.

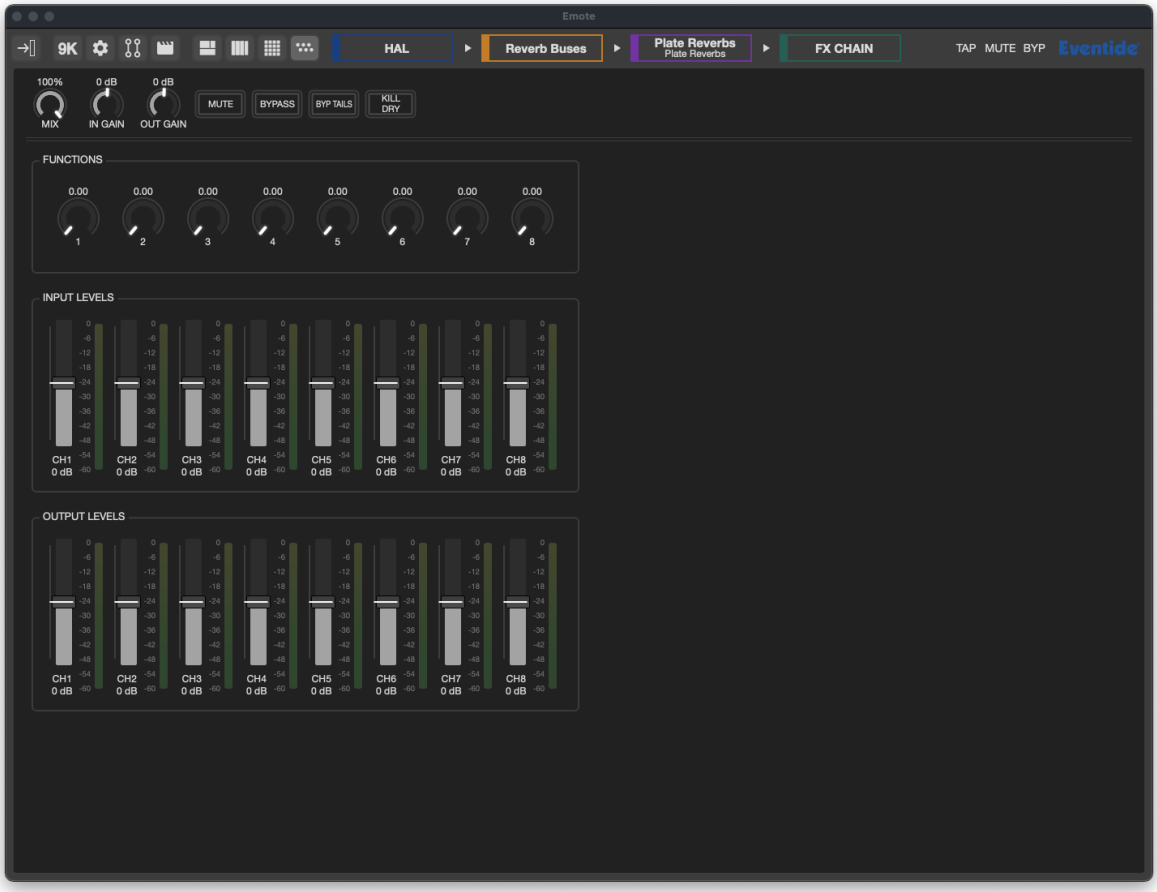


Fig. 3.5 **FX Chain Parameters View**

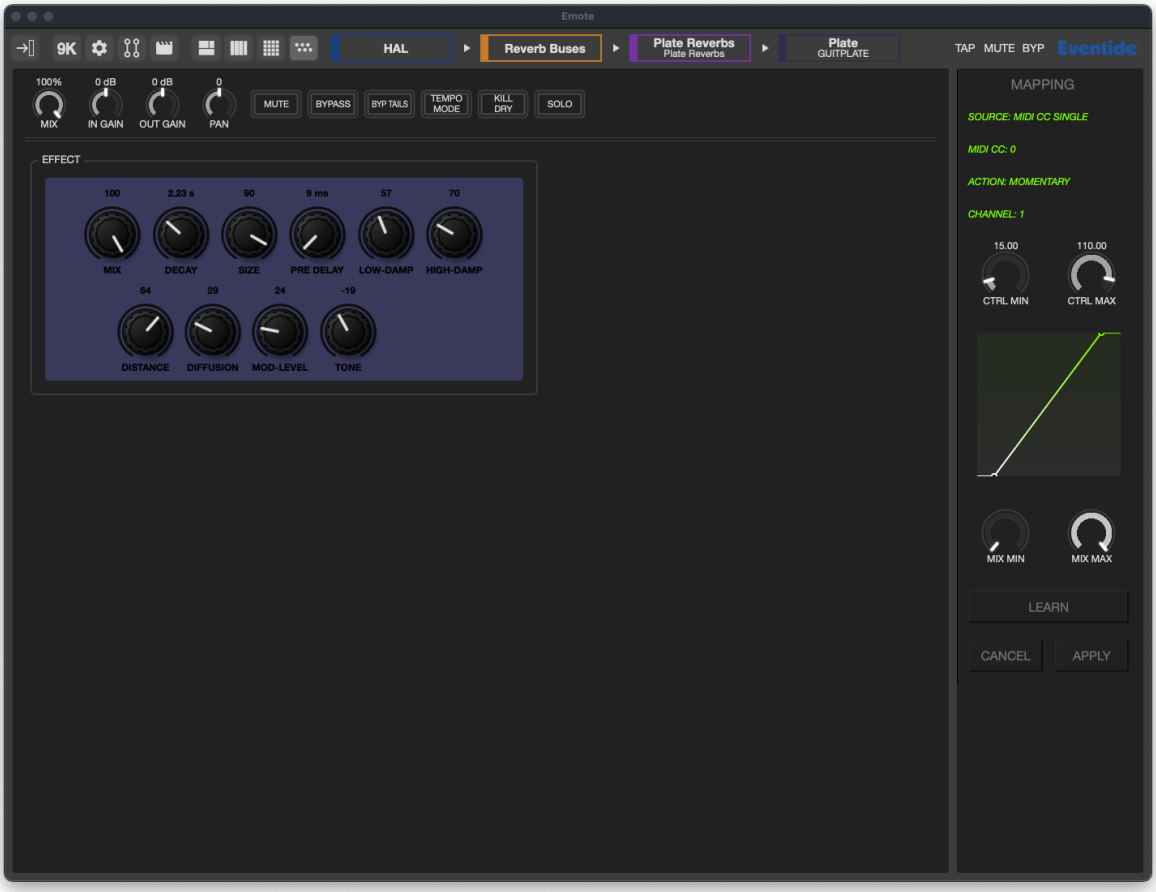


Fig. 3.6 **Algorithm Parameters View**

3.3.4.1. MIDI and Controller Mapping

Parameters and functions in Emote can be controlled by three different kinds of sources:

- MIDI (see appendix C)
- Expression (see section 2.1.7)
- Functions (see section 3.4.1)

This is done by right-clicking a specific parameter or function, then selecting `MAP`. Doing so will bring up the mapping window along the right-hand side, where you can select:

- Controller source
- Action type
- Control and parameter minimum/maximum ranges

You can also click the “Learn” button to assign a controller automatically. After pressing Learn, move the physical control you wish to pair with the parameter and click “Apply”.

After clicking Apply, the parameter will show a glowing green indicator across the range of the mapping.

3.3.5. Combined View

Combined View shows the FX Chain, Algorithm, and Parameter views all at once. Parameter View is located at the bottom, FX Chains View is located on the left, and Algorithm View is located on the right.

Parameter View displays the parameters of the algorithm or FX chain that is currently selected in Algorithm View, and Algorithm View displays the algorithms within the FX chain that is currently selected in FX chains View.

Fig. 3.7 **Combined View**

3.3.6. Scenes View

Scenes View allows you to create and edit Scene Maps and their contained lists of scenes. Scenes View has four sections:

1. The left side of the screen displays a list of Scene Maps.
2. The middle portion of the screen displays trigger sources and scenes contained in the currently active Scene Map.
3. The right side of the screen displays options for the currently highlighted scene or trigger source.
4. The bottom portion of the screen displays the FX chains in the currently loaded scene.

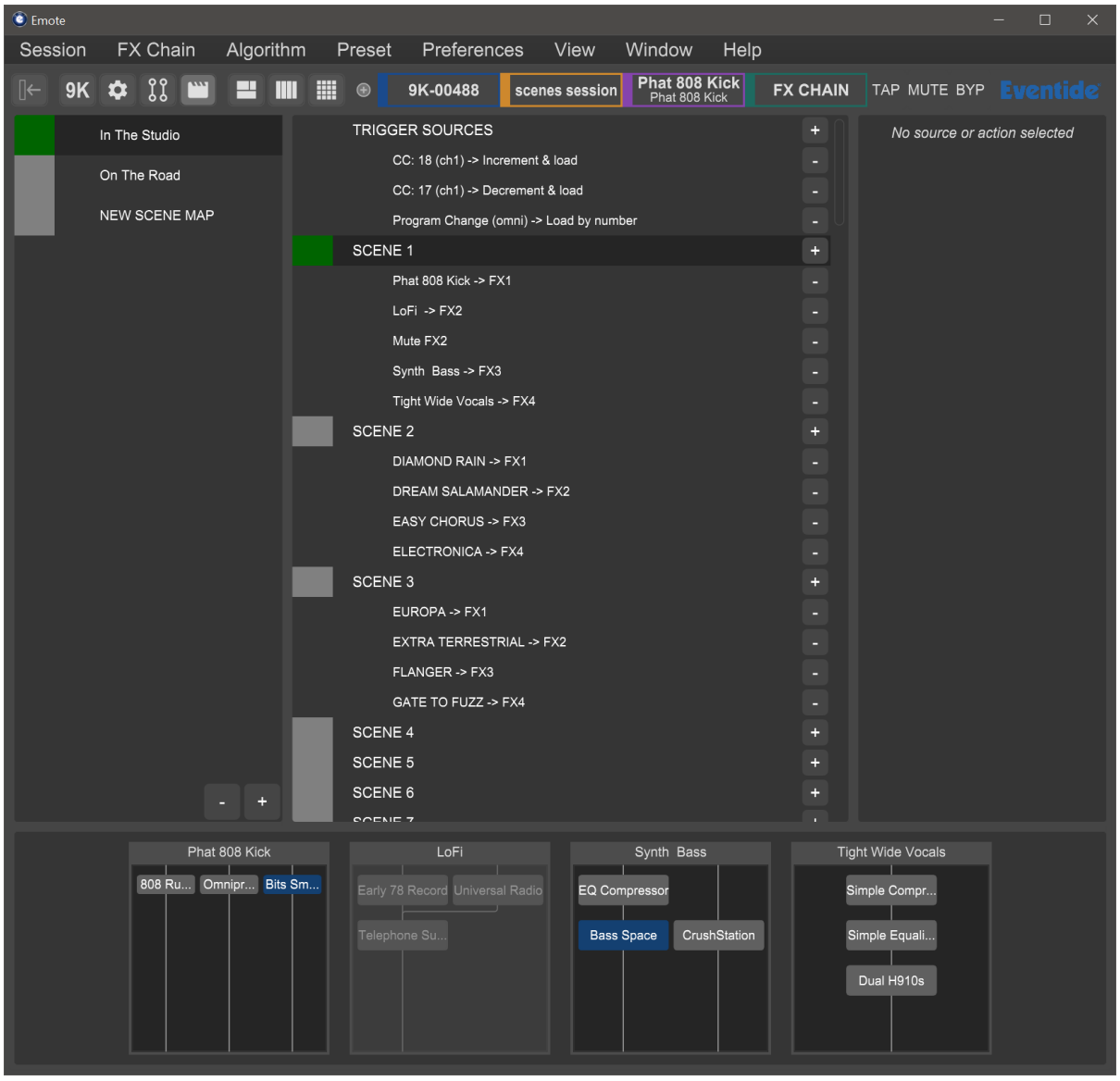


Fig. 3.8 **Scenes View**

Scene Maps

The Scene Maps portion of the left side of the screen will be your starting point for creating and using scenes. Each Scene Map can have a total of 128 scenes and only one Scene Map can be active at any given time.

- To create a Scene Map, press the “+” icon at the bottom of the Scene Map portion of the screen and a new map will be added to the list. To rename a Scene Map, right-click its text and a prompt to rename it will appear.

- To delete a Scene Map, highlight the map that you'd like to delete and press the "-" icon. A prompt will ask you to confirm that you'd like to delete this map.
- Double click a Scene Map or press `RETURN` to activate it. The square to the left of its text will change from grey to red, indicating that it is now active. Double-click it or press `RETURN` again to deactivate it.

Scenes and Trigger Sources

The middle and right portions of the screen work in tandem to provide all the information needed for creating and customizing your scenes.

The middle portion of the screen offers control of two key areas:

1. The list of trigger sources contained within the currently active Scene Map.
2. The list of scenes contained within the currently active Scene Map.

Creating and Editing Trigger Sources

- To add a new trigger source, click the "+" icon to the right of the "TRIGGER SOURCES" text.
- Highlight a trigger source, and a list of options will show up on the right side of the screen.
- You will now have the ability to customize your source mode, message type, and MIDI channel (refer to Section 4.8.3.1: Trigger Source Options for more information about specific trigger sources).
- Alternatively, click "Learn" and move your desired external control source to automatically map it as the trigger source.
- To delete a trigger source, click the "-" icon to the right of the trigger source that you'd like to delete.
- To delete all trigger sources, right-click the "TRIGGER SOURCES" text and a prompt will appear to delete all sources.

Creating and Editing Scenes

Scenes can be edited manually or by using the Scene Capture mode.

- To manually add a new action, press the "+" icon to the right of the scene that you'd like the action to be added to. A new action will appear below that scene.
- Highlight the newly added action and a list of options will appear on the right side of the screen.
- You will now have the ability to Load, Enable, Mute, or Bypass an FX chain or algorithm for any of the four FX chains.
- To capture a scene, right-click the scene and select the Capture Scene prompt.
- This will take a "Snapshot" of all FX chain settings and parameters in the current session.
- To delete all actions, right-click the scene and select the "Remove All Actions" prompt.
- You can give your Scenes custom names by right-clicking and selecting "Rename".

FX Chain Display

The bottom portion of the Scenes page in Emote will display the currently loaded FX chains and algorithms. This is useful for monitoring what changes are being made as you load scenes.

3.3.6.1. Working with Scenes in Emote

Scenes can be selected and activated by clicking on them in the Emote GUI or by sending the appropriate MIDI message as configured above.

- Double-click a Scene Map or press `RETURN` to activate it. You will now be able to select and activate any of its contained scenes.
- Double-click a scene or press `RETURN` to activate it. The square on the left side of its text will be turned yellow as it is pending, and then green once it has loaded.
- When sending "Increment Up/Down" MIDI messages, the square to the left of the scene's text will become yellow to indicate it is pending. When a "Load Current" message is sent the square will become green, indicating that it has been loaded.
- If the square to the left of a scene's text is red, then the Scene Map in which it is contained is not active and therefore will not accept MIDI messages.

Note

Scenes cannot be triggered by automating the Emote plugin parameters in a DAW. To trigger scenes from your DAW, send the appropriate MIDI messages via USB-B or a MIDI interface.

3.3.7. Direct I/O View

Direct I/O View functions similarly to steps 2 through 4 in Algorithms View, Configuring Audio Connections (Section 3.3.3.2)

Here, any input can be multed to multiple outputs, but multiple inputs cannot be summed to one output. Also keep in mind the limitations listed in section 4.1.1: Signal Routing.

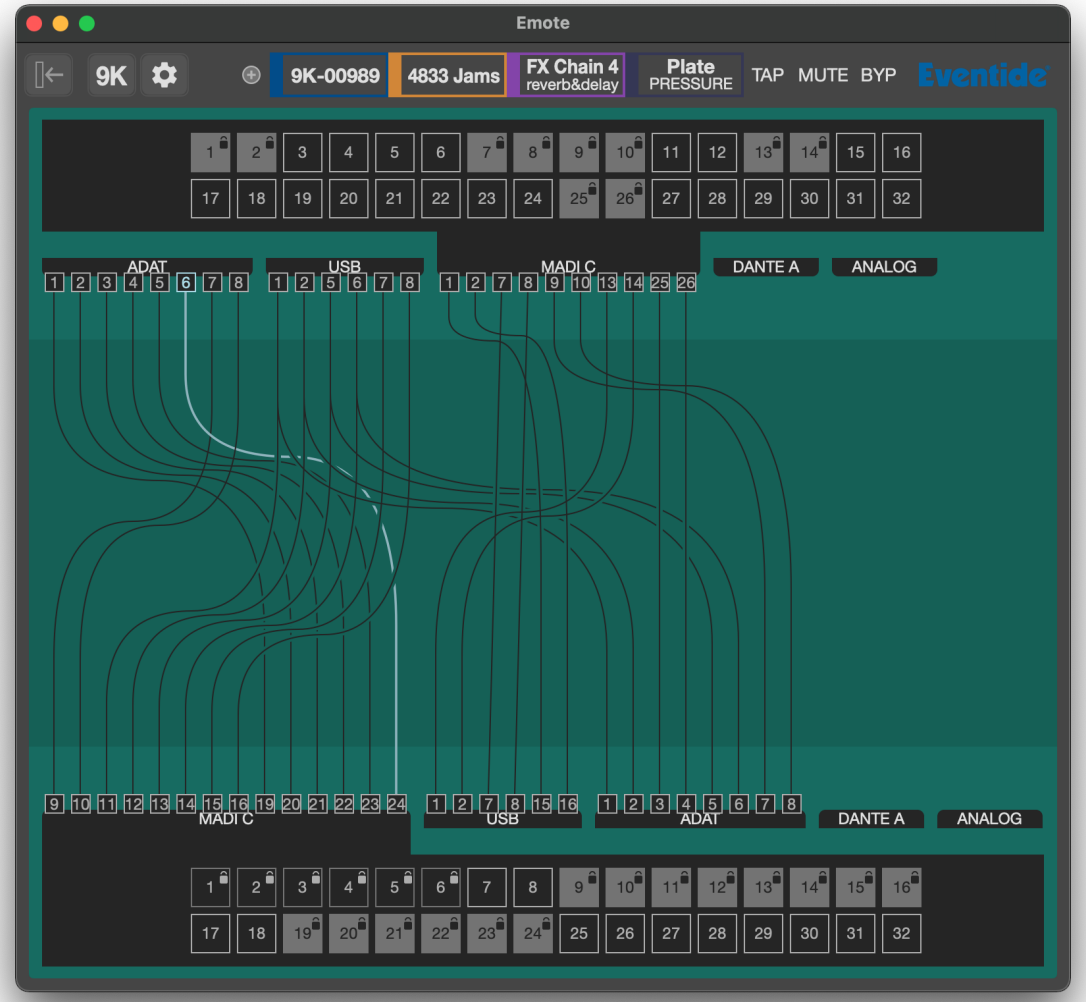


Fig. 3.9 **Direct I/O View**

Note

Direct I/O is not effected by the global mute.

3.3.8. Setup View

Please refer to Section 2.1: Setting up Your H9000 for information about these settings.

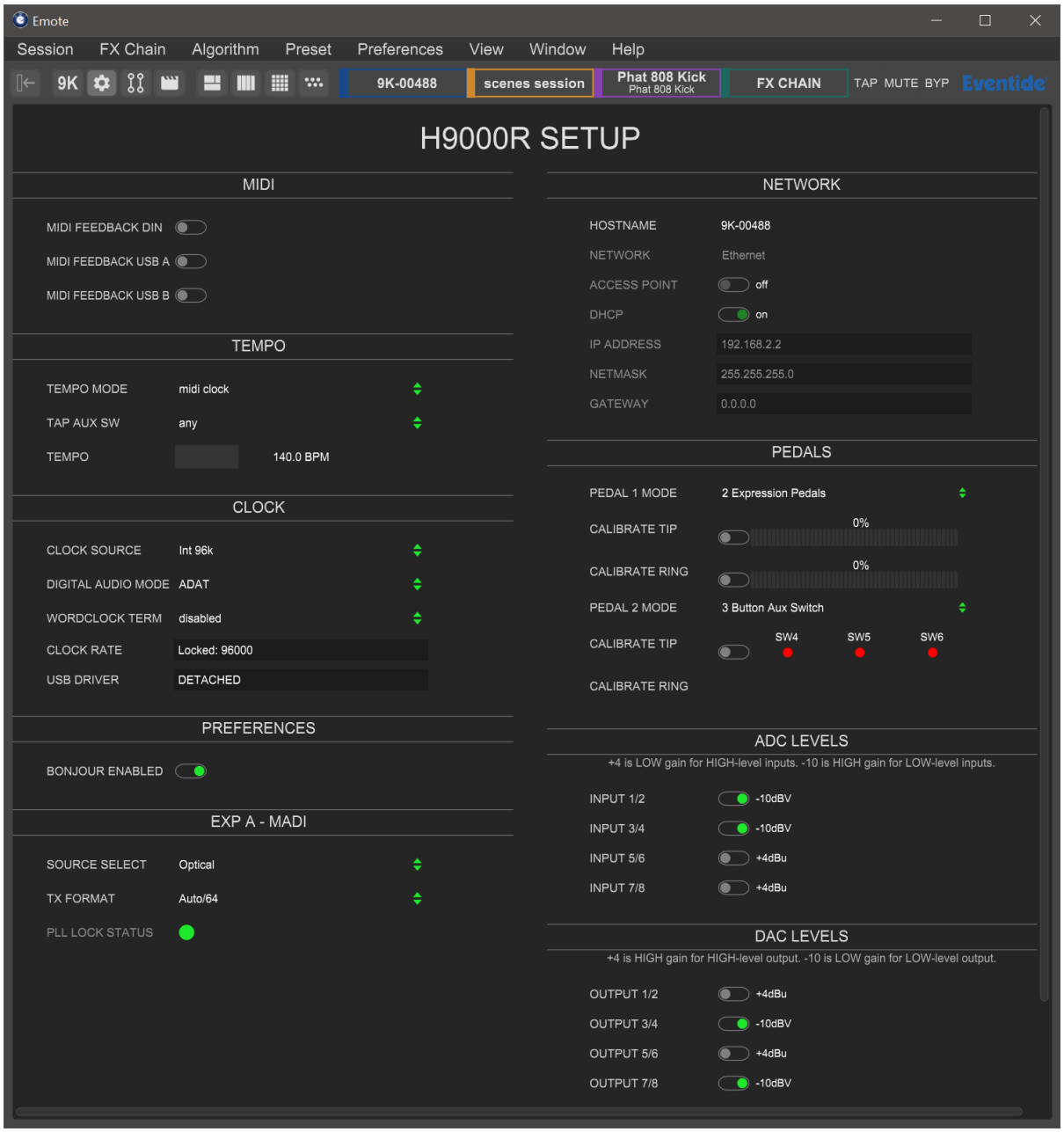


Fig. 3.10 **Setup View**

3.4. Using Emote as a Plug-in

Emote can be used in your DAW using VST3, VST, AU, and AAX plug-in formats. Only one instance of Emote is needed in a given DAW project, and it does not matter which audio track it is placed on.

> **Note**
>
> Using Emote as a plug-in does not automatically route audio connections to the H9000, all audio connections must be configured by using your selected DAW's method of routing external audio effects.

The following features are only available while using Emote as a plug-in:

3.4.1. Function Automation

Algorithms in the H9000 can contain any number of parameters; however, most DAW hosts assume that the plug-ins they are hosting will have a static number of parameters. For this reason, it is necessary to use logical parameters to map between what the DAW sees, and what the H9000 has currently loaded. This is achieved by using the H9000's Functions as the interface to the DAW automation. This approach also has the added benefit of being able to aggregate multiple H9000 parameters and automate them in a single automation lane.

When the plug-in version of Emote is loaded into your DAW, the options for automation will appear on the track that Emote is placed. Each of the four possible FX chains contain eight Functions, as well as the following options to be automated by the DAW:

- Mute
- Bypass
- Input Gain
- Output Gain
- Mix Controls

To begin automating some parameters:

- Follow the instructions in Section 3.3.4.1 MIDI and Controller Mapping to assign some H9000 parameters to a function.

- In your DAW, write some automation data to the lane corresponding to that specific H9000 function.

Note that while four FX chains are always exposed in the DAW, there may not actually be four chains running on your H9000. In this case, automating any unused ones won't have any effect.

If you replace an FX chain within the H9000, the DAW automation data will still exist, and will apply to whatever parameters are assigned to the Function in the newly created FX chain.

Warning

Because Function data is sent via the network, Functions will stutter if the WiFi is weak, or if the network is congested.

3.4.2. DAW State Recall

When Emote is present as a plug-in in your DAW project, your DAW will automatically save the current state of all of the H9000's settings within Emote. This includes settings for Sessions, FX Chains, Algorithms, Presets, and Scenes.

When you re-open the DAW project and connect to Emote, a pop-up message will display "This DAW project holds state data for 9K-XXXXX. Would you like to restore it?"

- Select *YES* to restore your H9000's settings for this DAW project. You will be prompted to save your current H9000 session before the H9000's state data from the DAW project is loaded.
- Select *NO* to keep your current H9000 session. If you later decide you'd like to restore the H9000's DAW state data, reload the DAW project *without* saving.

Note

While the DAW State Recall is a useful feature for saving your H9000 data, it is best practice to also save an H9000 session with all of your settings.

3.4.3. DAW Tempo Sync

The H9000 can automatically sync its tempo to your DAW project when Emote is used as a plug-in. To do this, set the H9000's Tempo Mode to "DAW" (Section 3.3.8).

4. FRONT PANEL OPERATION

In this section, we will show you how to navigate the H9000 front panel and operate the unit.

Note

This chapter is not necessary for the H9000R, which has no front panel controls.

4.1. Getting to know your H9000

4.1.1. Signal Routing

There are a few general “rules” about signal routing in the H9000 which are useful to keep in mind.

- You can only use one digital audio protocol at any given time. This will be determined in your Digital Audio Mode setup.
- When using S/PDIF over the optical ADAT connectors, you can only work at sample rates of 44.1kHz and 48kHz. Sample rates of 88.2kHz and 96kHz are not supported.
- FX chains support 32 inputs and 32 outputs. The outputs of an FX chain can be “multed” to an arbitrary number of physical Input/Output channels, computer audio interface channels, etc.

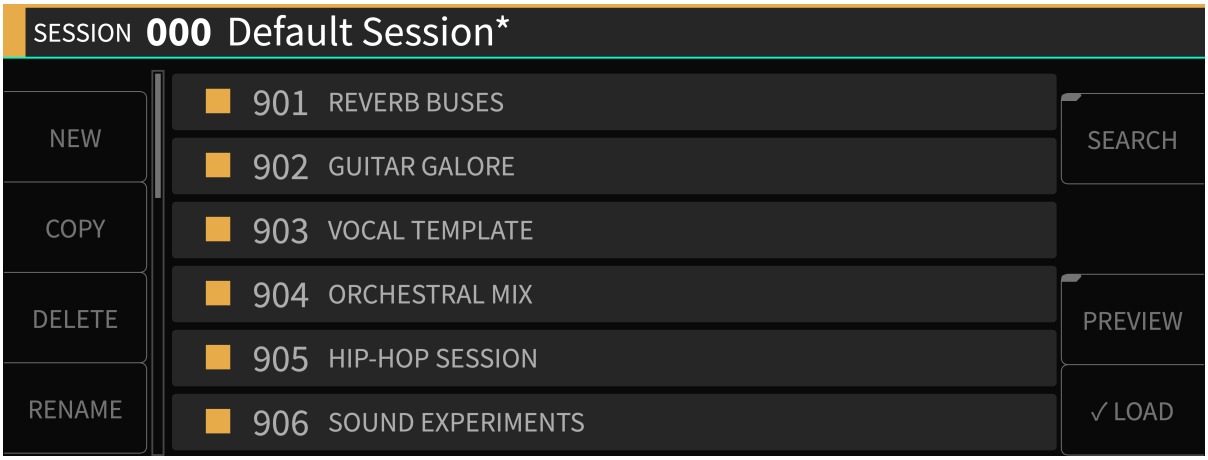
4.1.2. Soft Keys

The H9000 user interface has context-sensitive “soft keys” on the left and right sides of the display.

There are two different types of soft keys on the H9000:

- “Normal” soft keys: These execute their labeled action when pressed.
- “Latching” soft keys: When pressed, they change the front panel display to a different state. Think of the latching soft keys as a quick way to access “sub-menus” or alternate screens on the H9000. Latching soft keys are visually distinguished by a small triangular “flag” in the top corner of the key.

4.2. Sessions



A *Session* is the "big picture" for the H9000, a snapshot of the current state of all settings on the unit.

To load a session, press the front panel *Sessions* button; the button will illuminate to indicate that you are currently working with the *Sessions* view screen.

The screen will switch to the *Sessions* view, presenting you with a numbered list of sessions available for loading. The presently loaded session is displayed in the title bar at the top of the screen.

There are several different ways to navigate through the list of sessions:

- Press the cursor up/down buttons or rotate the wheel to scroll through the list one entry at a time.
- Press the cursor left/right buttons to navigate through the sessions list one page at a time.
- The session you have currently navigated will highlight in green.
- Press the *Enter* button to load the highlighted session.
- The chosen session will load and you will automatically be taken to the *FX Chains* screen where you can edit the session's FX chains.

4.2.1. Session Screen Soft Keys

The eight *Sessions* view soft keys offer additional session-related functions:

New

Press the *New* soft key to create a new session. If you have made changes to the currently loaded session, you will be prompted to save your changes first. You will then see the *Create New Session* screen. Press the *Create New Session* soft key (or the *Cancel* soft key if you change your mind). The new session you have created will be titled “Default Session” and will contain a default FX chain.

Copy

Press the *Copy* soft key to make a copy of the currently highlighted session. The naming screen will appear, allowing you to create a new name for the copy of the session. This is similar to the “Save As” function on a word processor application, where you can save a copy under a different name.

Delete

Press the *Delete* soft key to delete the currently highlighted session in the list. You will be prompted with a confirmation screen; press either the *Cancel* or *OK* soft keys. Note that you cannot delete factory sessions.

Rename

Press the *Rename* soft key to rename a currently highlighted session. You will then be presented with the naming screen where you can enter a new name. Note that you cannot rename factory sessions. If you attempt to do so, you will be offered the option of creating a new user session, which you can then rename.

Load

Press the *Load* soft key to load the currently highlighted session. This soft key duplicates the function of pressing the *Enter* key when a session is highlighted.

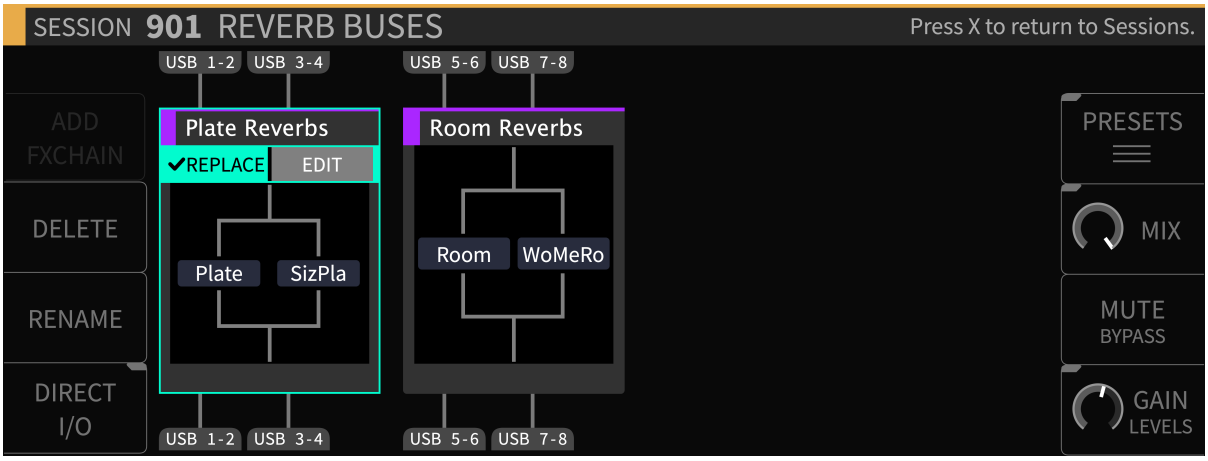
Search

Press the latching *Search* soft key to bring up the search screen, where you can use the keypad to type a keyword. Sessions containing the letters you type will appear in the filtered list of available sessions.

Preview

Press the latching *Preview* soft key to get a visual representation of the layout of the session, which includes the number of FX chains, the algorithms in each FX chain, and the physical inputs and outputs. Since loading a session is not an immediate operation, the *Preview* function offers a quick way to see what the session contains before you actually load it.

4.3. FX Chains



Press the front panel *FX Chains* button to switch the display to the *FX Chains* view screen, where you can view and edit the FX chains of the currently loaded session.

Press the cursor left/right buttons or use the wheel to highlight the different FX chains. Press the cursor up/down buttons to move through the signal chain blocks in the currently highlighted FX chains block. An FX chain is comprised of audio inputs, the algorithms contained in the FX chain, and audio outputs.

4.3.1. Configuring the Audio I/O of your FX Chain

Use the cursor buttons to highlight the *Edit* block of the FX chain's audio input section; you will then be presented with choices for selecting both the physical input connector, and the channel number of your selected input connector. Use the wheel to scroll through the available choices. When you have found your desired audio input, press the *Enter* key. Your chosen input source will now be routed to the input of the FX chain and the display will show a connecting line.

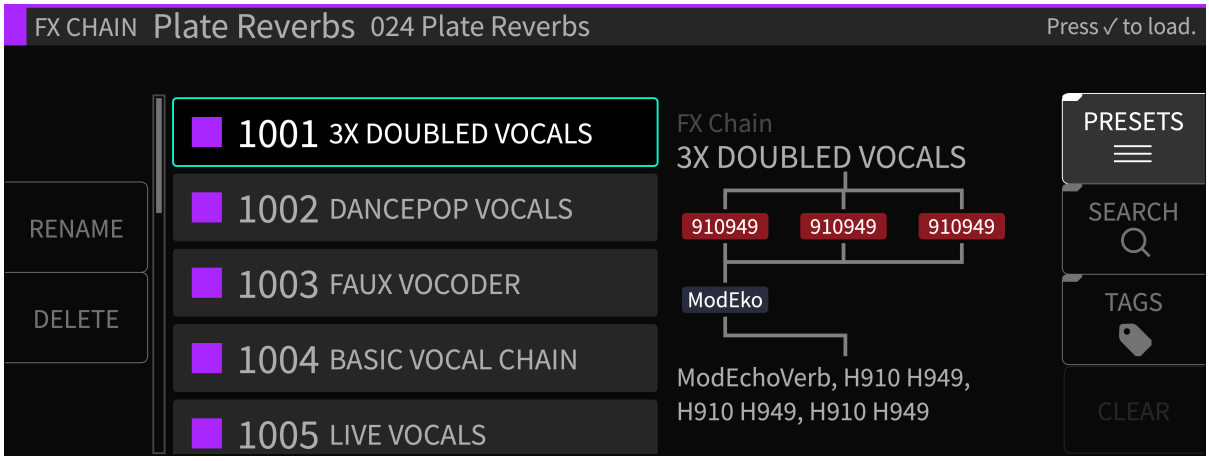
To assign an output to the FX chain, follow the same procedure using the *Edit* block located at the bottom of the *FX Chain* display.

Note that you cannot select an output that has been assigned to another FX chain.

4.3.2. Replacing an FX Chain

The middle section of the FX chain block shows the algorithm(s) that make up the current FX chain. To replace the current FX chain with a different FX chain preset:

- Highlight the *Replace* block and press the *Enter* key. You will be presented with a list of available FX chain presets.
- Navigate to the FX chain preset you wish to use as a replacement. Press the *Enter* key to insert the chosen preset into the FX chain.



At any time, you can press the *Cancel* button to back out of the current screen state and return to the previous state. Pressing the *Cancel* button a final time will exit the *FX Chains* screen and return you to the *Sessions* screen. You can always return to the *Sessions* screen by pressing the *Cancel* button once for each iteration of menus you have entered. In other words, if you're in too deep and need to reset, press the *Cancel* button to get back home.

4.3.3. FX Chain Soft Keys

The *FX Chains* screen contains eight soft keys that offer the following functions:

Add

Press the *Add FX Chain* soft key to add a new FX chain to the currently loaded session. You can have up to four FX chains in your H9000. The H9000 will prompt you to select a name and I/O configuration for the FX chain, which will be routed automatically upon creation.

Delete

Press the *Delete* soft key to delete the highlighted FX chain. Note that you cannot delete the FX chain if it is currently the only one loaded in the session.

Rename

Press the *Rename* soft key to bring up the naming screen, where you can rename the currently highlighted FX chain container in the session. The name of the FX chain *container* is distinct from the name of the FX chain preset *loaded into it*; this can be useful for distinguishing an FX chain's purpose (e.g., Guitar 1, Guitar 2) if the same preset is loaded into multiple containers.

Direct I/O

Press the *Direct I/O* latching soft key to view the "Direct I/O" screen. Direct I/O lets you use your H9000 as a virtual patchbay; any input can be routed to any output. This lets you take advantage of the H9000's pristine analog converters and complex routing capabilities, even when you do not require any of the signal processing power that the H9000 can provide. Note that this is not an FX chain, but rather a direct audio pathway between a physical audio input and output. Direct I/O assignments are saved within Sessions, and are not effected by the global mute button.

In the *Direct I/O* screen, use the cursor keys to highlight the input/output you wish to assign, then use the wheel to select the physical input/output you wish to use in your direct I/O pathway.

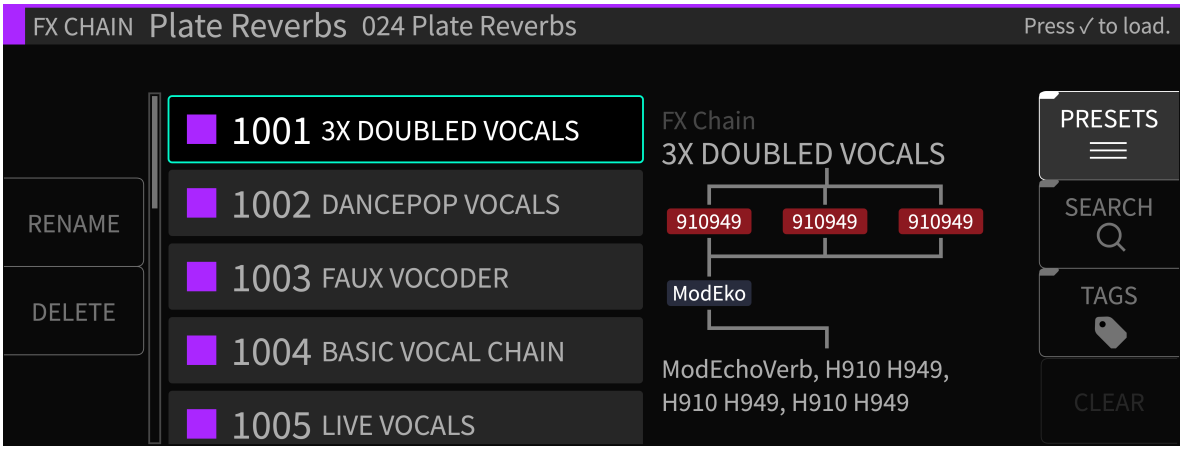
Additional soft keys are available in the *Direct I/O* screen:

- Add: Press the *Add* soft key to create a direct I/O pathway within the *Direct I/O* screen. Pressing the *Add* soft key a second time will connect that pathway and add another new pathway.
- Connect: Press the *Connect* soft key to create a connection between a direct I/O pathway that has not yet been connected.

- Disconnect: Press the *Disconnect* soft key to delete a direct I/O pathway that you have highlighted. Multiple pathways can be disconnected at once by highlighting the I/O source and pressing the *Disconnect* soft key.
- Page Left/Right: Press the *Page Left/Right* soft keys to navigate the different direct I/O pathways you have created when there are more of them than fit on the screen at one time.

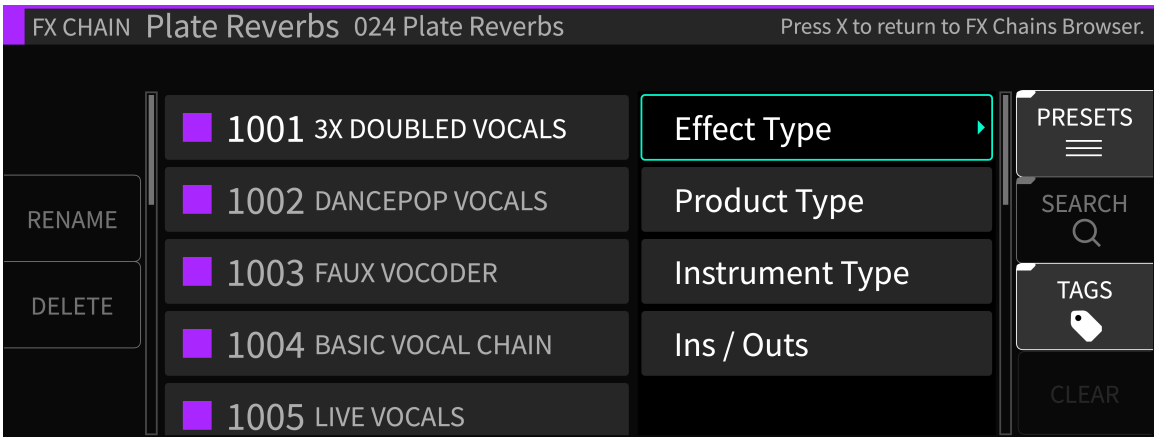
Presets

Press the *Presets* latching soft key to enter the FX chain preset browser. Navigate to the FX chain you wish to load and press the *Enter* key.



Additional soft keys are available in the *Presets* browser screen:

- Rename: Press the *Rename* soft key to rename the currently highlighted FX chain. Note that factory FX chains cannot be renamed.
- Delete: Press the *Delete* soft key to delete the currently highlighted FX chain preset. Note that factory FX chains cannot be deleted.
- Search: Press the latching *Search* soft key to bring up the virtual keypad, where you can enter in letters and numbers to search for a FX chain name. As you enter in more characters, the FX chain list is filtered to show only FX chains that contain the characters you are typing.
- Tags: Press the latching *Tags* soft key to switch the right-hand side of the display to the Tags screen. You will be presented with a list of tag categories. Use the cursor keys and enter key to select.



The following tag categories are included: Effect Type, Product Type, Instrument Type, and Ins/Outs.

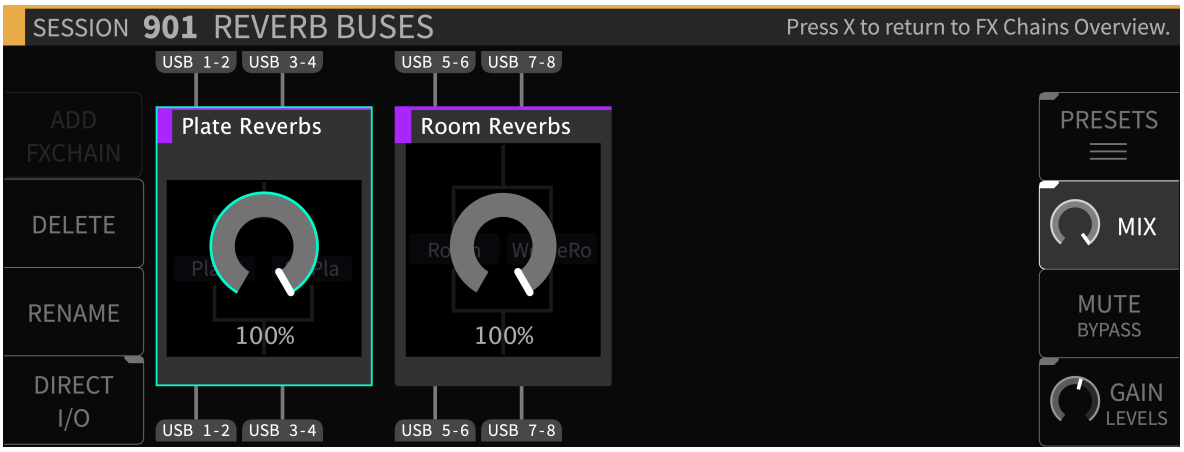
Once you have selected a tag category, you will be presented with a menu of tags within that category. Use the cursor and enter keys to select a tag. It will be added to the current tag list and be shown above the tag list. Note that you can select as many tags at once as you like. As you add tags, the main list of FX chains will be filtered to only show presets that correspond to the tag(s) you have selected.

Many of the algorithms on the H9000 will carry multiple tags. For example, Blackhole will fall under the tags of Product Type H9 as well as Effect Type Reverb. This powerful feature is a great way to narrow down the H9000's vast list of FX chains to a more manageable selection that is appropriate to the task at hand.

- Clear: Press the *Clear* soft key to clear all currently loaded tags and return the main list of FX chains to its untagged state.

Mix

Press the latching *Mix* soft key to adjust the mix levels of the FX chains present in the currently loaded session. The display will switch to the *Mix* screen, showing level controls for each FX chain. Use the cursor keys to navigate to the FX chain for which you wish to adjust the level, then use the wheel to adjust its mix level.



Mute/Bypass

Press the *Mute/Bypass* soft key to mute the currently highlighted FX chain. Press it again to unmute the FX chain. When muted, a red FX MUTE label will appear at the bottom of the FX chain block.

Long-press the *Mute/Bypass* soft key to bypass the currently highlighted FX chain. Long-press it again to un-bypass the FX chain. When bypassed, a red FX BYPASS label will appear at the bottom of the FX chain block.

Gain/Levels

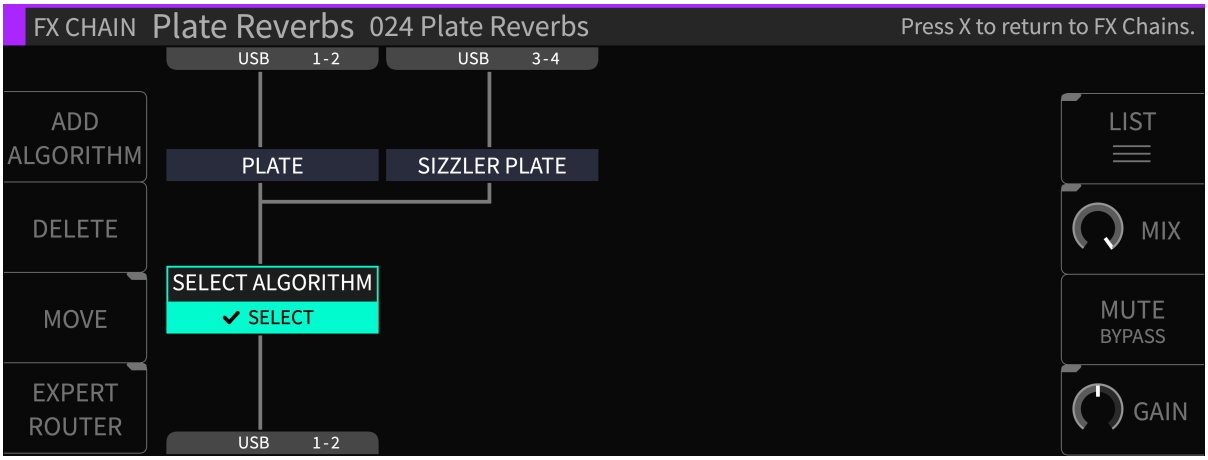
Short-press the latching *Gain/Levels* soft key to adjust the output gain levels of the FX chains present in the currently loaded session. Similar to the *Mix* mode, the display will show level controls for each FX chain. Use the cursor keys to navigate to the FX chain for which you wish to adjust the level, and then use the wheel to adjust the output gain.

Long-press the latching *Gain/Levels* soft key to show signal levels for the audio inputs/outputs of the currently highlighted FX chain.

Additional soft keys are available in the *Gains/Levels* screen:

- Control Map: Press the latching *Control Map* soft key to view external control mapping options for Gains/Levels. Refer to Section 4.5: Parameters for more information about external controller mapping.
- Clear Clip: Press the *Clear Clip* soft key to clear any overloads that are displayed on the I/O meters.
- Group: Press the latching *Group* soft key to group together the inputs or outputs of the selected FX chain input or output level. When the latching soft key is illuminated, adjusting the level of either group's fader will adjust the level all faders in each grouping.

4.4. Algorithms



In the H9000, each FX chain can contain up to four separate signal processing algorithms. The *Algorithms* screen is where you work with these algorithms; you can navigate to it by:

- Pressing the *Algorithms* button on the front panel.
- Highlighting the *Edit* button in the center portion of an FX chain block, then pressing the *Enter* button.

The screen will switch to the *Algorithms* display and the front panel Algorithms button will illuminate.

4.4.1. Algorithms Overview

The *Algorithms* overview screen displays the algorithms that make up the currently highlighted FX chain. The display shows the different algorithms present, their routing within the FX chain, and the audio inputs/outputs that feed the FX chain.

Rotate the wheel to navigate to the different objects in the FX chain, which include:

- Algorithm processing blocks
- Input and Output blocks

When viewing the *Algorithms* screen, pressing the *Enter* button works contextually for whatever object you have highlighted:

- *Algorithm*: When the *Edit* label is highlighted, pressing *Enter* takes you to the *Parameters* view page (see the Parameters section for more details) When the *Replace* label is highlighted, pressing *Enter* takes you to the main list of algorithms to choose from.
- *I/O Label:* Pressing *Enter* takes you to the editor screen for the audio inputs/outputs.

4.4.2. Configuring Audio I/O For an Algorithm

To configure the audio I/O on the *Algorithms* view screen:

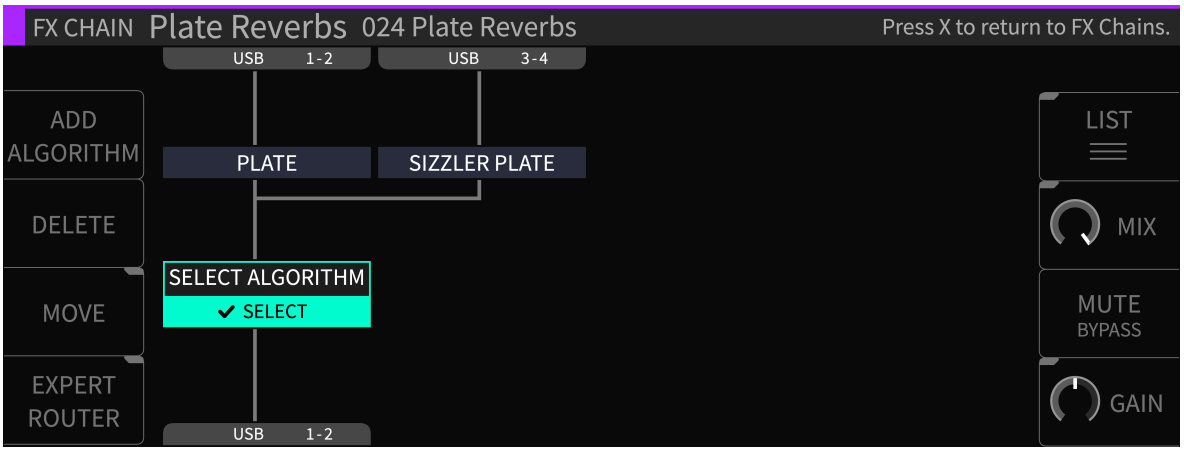
- Use the cursor keys to highlight the audio input or output block you wish to configure, and press the *Enter* key.
- You can then use the cursor keys to highlight the audio I/O category (Analog, Digital, USB, Expansion Card), then cursor over to the channel number and adjust it separately. This type of fine-tuned I/O control is particularly important when an expansion card is present, which can have 32 I/O channels per slot.
- Use the wheel to cycle through the available input or output assignments.

4.4.3. Algorithm Screen Soft Keys

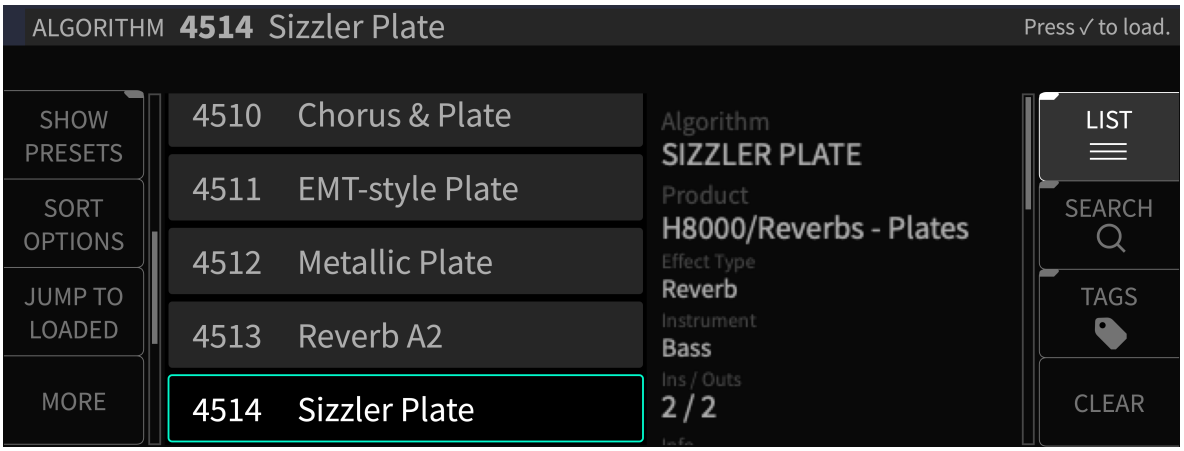
The *Algorithms* view screen contains eight Soft Keys:

Add Algorithm

- Press the *Add Algorithm* soft key to add a new algorithm to the currently selected FX chain. A new algorithm block will appear labeled “Select Algorithm”.
- Use the cursor keys to highlight the “Select Algorithm” block and press the *Enter* key.



- Use the navigation controls to select an algorithm. Each entry contains helpful info (Ins/Outs, a description of the algorithm, tips on its use, etc.).



Delete

Press the *Delete* soft key to remove the highlighted algorithm from the FX chain.

Move

Press the latching *Move* soft key to change the location of an algorithm within the FX chain. You can move the algorithm in an FX chain to be above or below any other algorithm in series, or next to another algorithm in parallel.

Expert Router

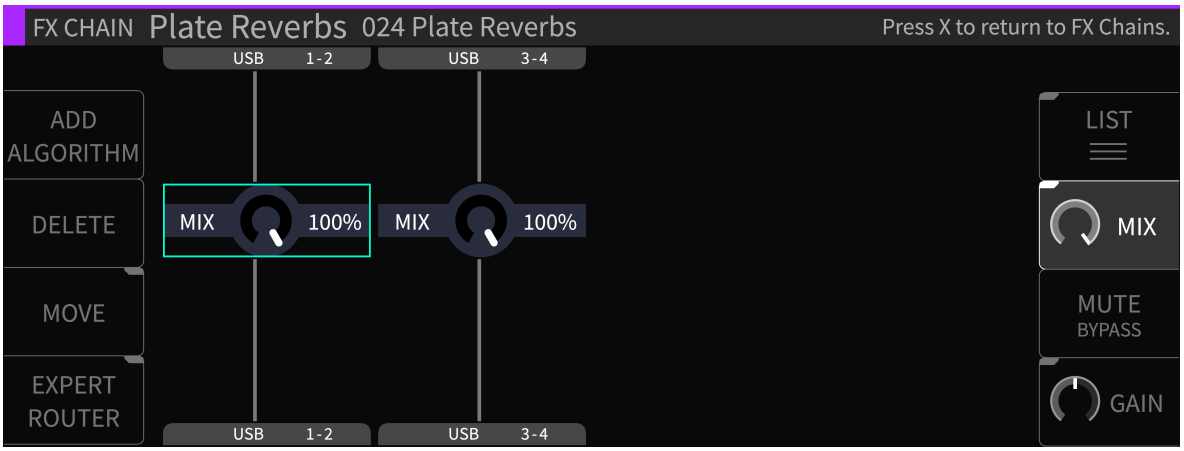
The H9000 allows not only serial routing of algorithms, but also parallel and mixed serial/parallel routing too. The expert router page is where you can view and control this aspect of the routing. Press the latching *Expert Router* soft key. The display switches to a graphical routing matrix, where you can view and edit a high level overview of all of the algorithms present in the current FX chains, their audio I/O assignments, and their signal routing configuration.

To make adjustments in the *Expert Router* screen:

- Use the cursor keys to highlight the aspect of the block diagram you wish to adjust (audio I/O block, routing, Algorithm).
- Use the wheel to adjust a highlighted audio I/O routing.
- Use the *Move* soft key to adjust a routing pathway when it is highlighted.

Mix

Press the latching *Mix* soft key to change the display to a set of mixing controls, which adjust the relative level of each algorithm within the FX chain. Navigate to the algorithm whose mix you wish to adjust, then rotate the wheel to change it.



Mute/Bypass

Press the *Mute/Bypass* soft key to mute the audio output of the currently highlighted algorithm. A red "M" label will appear at the right of the algorithm block when muted. Press the *Mute/Bypass* soft key again to unmute the audio output.

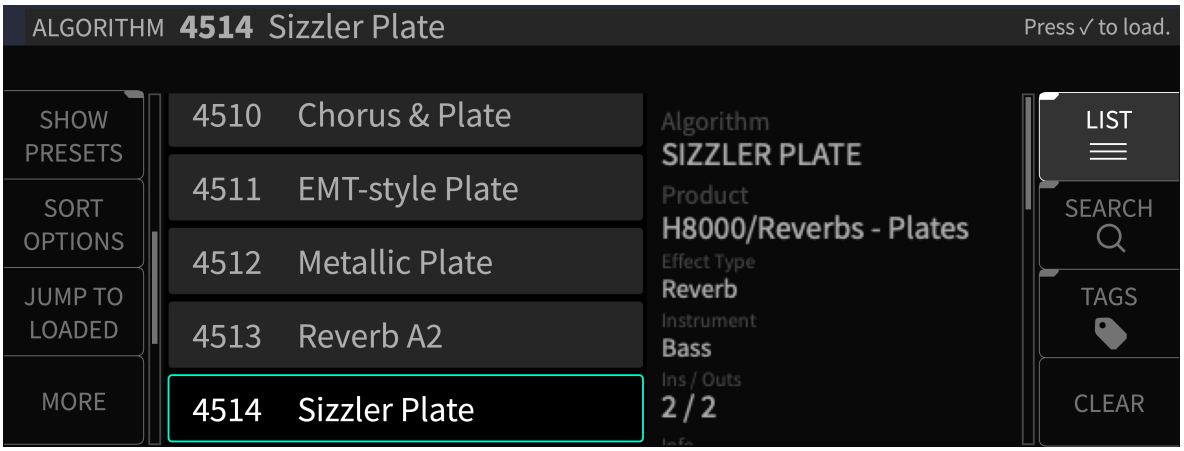
Long-press the *Mute/Bypass* soft key to bypass the currently highlighted algorithm. A red "B" label will appear at the right of the algorithm block when bypassed. Long-press the *Mute/Bypass* soft key again to un-bypass the audio output.

Gain

Press the latching *Gain* soft key to change the display to a set of gain controls which adjust the output level of each algorithm within the FX chain. The difference between the Gain control and Mix control is that Gain determines the overall signal presence of an algorithm in a FX chain, whereas Mix controls the overall Wet/Dry mix of an individual algorithm and its signal.

List

Press the latching *List* soft key to view the main list of all algorithms available on the H9000.



To the right of the list is a description field that shows information for the currently highlighted algorithm, including the name, the product from which it is sourced, the numbers of inputs and outputs available, and a description of the algorithm. Note that the description field will usually contain more text than can fit on the display; in this case, use the cursor keys to highlight the description field, then rotate the wheel to view the remaining text. The *List* screen the contains following soft keys:

- Show Presets: Some algorithms also contain *Presets*, which are variations of each algorithm with their parameters configured for different sounds. Press the latching *Show Presets* soft key to show/hide the presets contained within each algorithm.

- Sort Options: Press the *Sort Options* soft key to select which way the algorithms are sorted, including ID Ascending, ID Descending, Name, Recently Used, and Most Used. Use the cursor up/down keys to navigate to the sort option you wish to use, and then press the *Enter* key.

- Jump to Loaded: Press this soft key to jump to the algorithm that is currently loaded in the FX chain.

- Search: Press the latching *Search* soft key to bring up the virtual keypad and enter letters and numbers to search for an algorithm name. As you enter more characters, the list is filtered to show only algorithms that contain the characters you have typed.

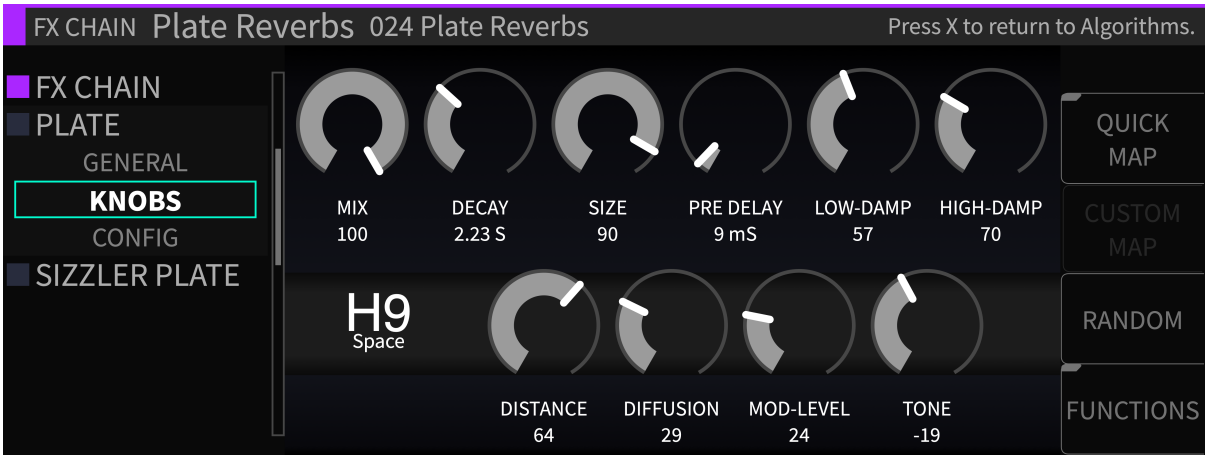
- Tags: Press the latching *Tags* soft key to switch the right-hand side of the display to the *Tags* screen. You will be presented with a list of tag categories, which you can select with the cursor keys and enter key. Tag categories are: Effect Type, Product Type, Instrument Type, and Ins/Outs.

  Once you have selected a tag category, you will be presented with a menu of tags within that category. Use the cursor keys and *Enter* key to select a tag; it will then be added to the current tag list and be shown above the tag list. You can select as many tags at once as you like.

As you add tags, the main list of algorithms will be filtered to only show algorithms that contain text matching the tag(s) you have selected. This powerful feature is a great way to narrow down the H9000's vast list of algorithms to a more manageable selection, one that is appropriate for the task at hand.

- Clear: Press the *Clear* soft key to clear all currently selected tags and return the main list of algorithms to its unfiltered state.
- More: Press the *More* soft key repeatedly to switch the soft keys above it to alternate functions. These additional soft keys include:
  - Jump to Last: Press this soft key to jump to the previously loaded algorithm.
  - Bank Up/Bank Down: Press these two soft keys to quickly jump to the next/previous bank of algorithms. The H9000 algorithms are organized into different banks, each one representing different categories of effects. Each bank begins with a new 3-digit number (100, 200, etc.)
  - Rename: Press the *Rename* soft key to rename the currently highlighted algorithm. Factory algorithms cannot be renamed.
  - Delete: Press the *Delete* soft key to delete the currently highlighted algorithm. Factory algorithms cannot be deleted.

4.5. Parameters



Each algorithm contains a set of parameters; you adjust these parameters to adjust the sound of the algorithm you have selected.

For example, a reverb algorithm contains parameters for settings such as room size, decay time, pre-delay, and so on. The *Parameters* screen allows you to view and adjust these parameters.

To switch to the *Parameters* screen:

- When the *Edit* label is highlighted, press the *Enter* key when an algorithm is highlighted. You will be taken to the *Parameters* screen for that algorithm.
- Alternatively, press the front panel *Parameters* button.

The *Parameters* view page offers a handy list on the left-hand side of the screen, which displays:

- The general parameters for the current FX chain container.
- The algorithms currently loaded in the selected FX chain.
- The parameter categories within each algorithm.

Use the cursor keys to highlight the parameter category for which you wish to view the parameters for. The screen will then display the parameters for only that category. Many algorithms contain a lot of parameters, and a single list could be cumbersome.

4.5.1. Editing Parameters in an Algorithm

To adjust a specific parameter:

- Use the cursor up/down keys to navigate to a parameter category. The right-hand side of the screen will display the parameters within that category.
- Use the cursor keys to highlight the parameter you wish to adjust.
- Use the *Enter* key and/or wheel to adjust the parameter you have highlighted. Note that you can hold the *Enter* key while scrolling to make fine-tuned parameter adjustments.

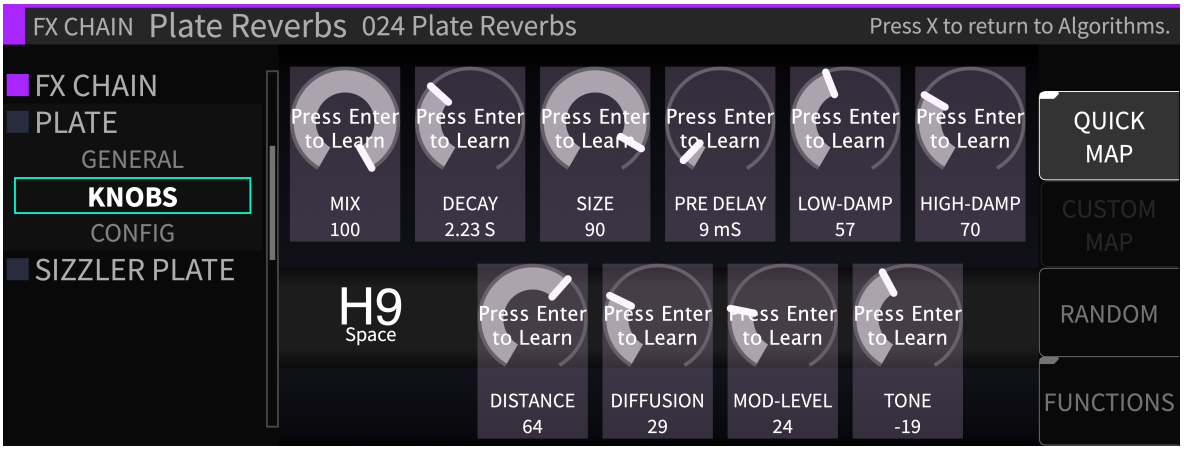
Note

You can also use the two middle-left soft keys to page up/down among the parameter categories, or the two outer-left soft keys to page up/down through the algorithms, while highlighting parameters within the parameter view. This can be a faster way to work than using the cursor keys exclusively; you can jump around the screen quickly and don't need to constantly "cursor left" back to the parameter category list.

The parameters screen contains four additional soft keys:

Learn

Press the latching *Learn* soft key to quickly map any of the parameters you are viewing.



The learn feature is extremely powerful, as it lets you quickly map a wide variety of external controllers to any algorithm parameter. Different aspects of the effects can be adjusted in real time using MIDI controllers, expression pedals, footswitches, control voltages, etc.

The MIDI functionality on the H9000 is “plug and play”. You can connect a MIDI device to the 5-pin MIDI-in DIN port, or connect a MIDI class compliant interface to one of the USB connectors. All MIDI Channel configuration is done when individual controllers are mapped, as opposed to in a central MIDI settings screen.

You can map multiple parameters to a single external control source; you cannot, however, have multiple control sources affect a single parameter.

For more complex mapping of multiple parameters to a single control source, please refer to the Functions section of this user manual.

The *Learn* feature works as follows:

- Press the *Learn* soft key; you will see a “Press Enter to Learn” label superimposed over each of the parameters on the screen.
- Use the cursor buttons to navigate to the parameter that you wish to map a controller to.
- Press the *Enter* button; the highlighted parameter’s label will change to “Move Any Controller”.

- Move the external control that you wish to pair to the highlighted parameter.
- The external controller and the parameter are now paired.

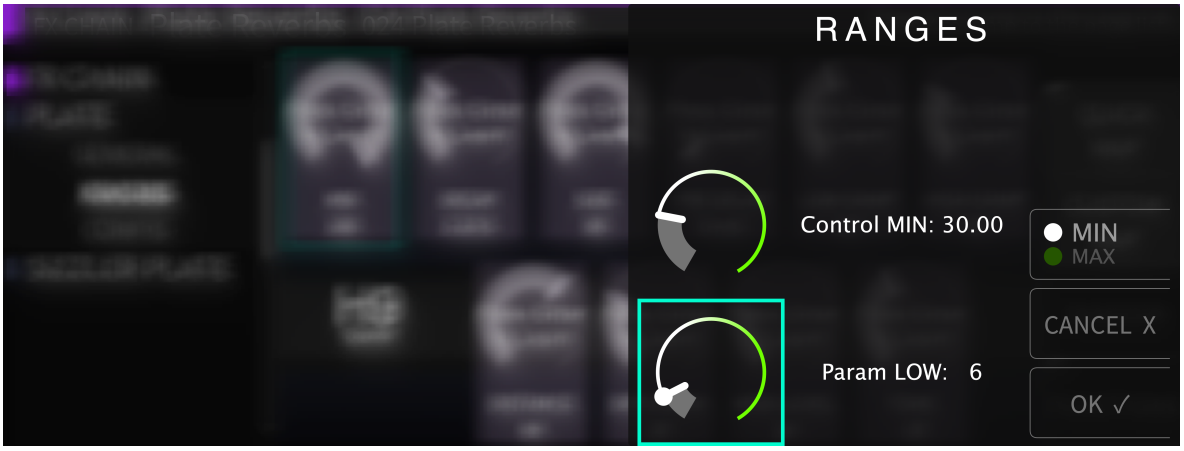
Custom Map

The *Custom Map* feature is also designed to map controllers to H9000 parameters, but with a much finer level of control. It works as follows:

- Use the cursor buttons to navigate to the parameter you wish to custom map a controller to, then press the *Custom Map* soft key.
- The "Controller Type" pop-up menu will appear, where you can select from the following mapping options:

  - Expression Pedal 1-4
  - Switch 1-6
  - Note On
  - Poly Pressure
  - MIDI CC Single
  - MIDI CC Double
  - MIDI Program Change
  - Channel Pressure
  - Pitch Wheel
  - MIDI Start
  - MIDI Stop
  - Function 1-8
  - Off

- Navigate to the controller type you wish to use then press the *Enter* key.
- You will then be presented with an additional set of controls that allow you to refine the mapping: Control Minimum, Control Maximum, Value Minimum, and Value Maximum.



- Use the cursor keys to navigate between the Control Range and Parameter range controls.

- Press the *MIN/MAX* soft key to toggle between adjusting the minimum or maximum value of a parameter, then use the wheel to set its value.

  Note that it is possible to change the direction of an external control source by inverting its min/max values. For example, if you set the Max value to 0 and the Min value to 100, when you move your external control source the mapped parameter will move in the opposite direction.

- When you have finished, press the *OK* soft key and the custom mapping will be completed.

- The display will return to the *Parameters* display. You will now see a green text label above the parameter, showing you the specific controller you have mapped to that parameter.

- To delete a custom mapping, use the cursor to navigate to the parameter mapping that you would like to delete, and press the *Custom Map* soft key. In the Controller Type menu, select "Off" and the mapping will be deleted.

Random

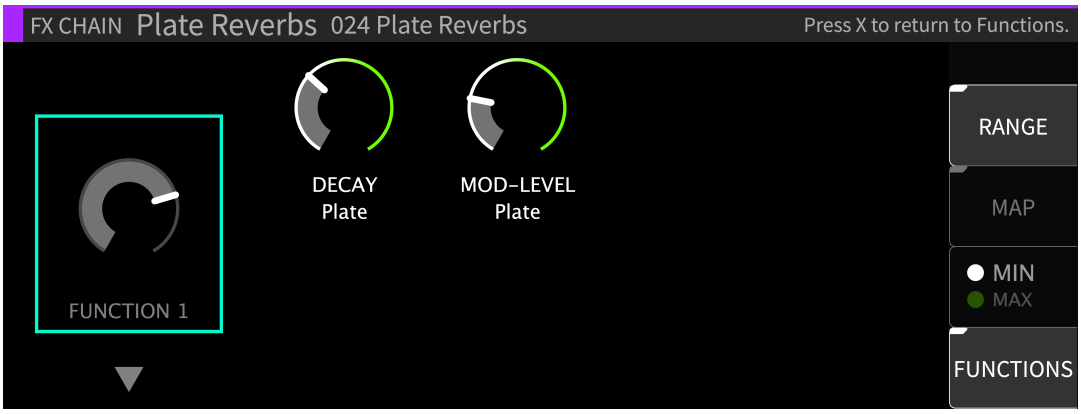
Press the *Random* soft key to assign random values to each and every parameter of the current algorithm. This creative function allows you to try out different algorithms in a free-form "what-if" type scenario, because the different parameter values are all selected with no specific purpose or sound in mind. Give it a whirl with your favorite algorithm and see what happens…

Functions

Functions in the H9000 act much like Macros in Ableton Live; they are groupings of parameters that can be adjusted with a single knob. There are eight Functions available in each FX chain.

Every function has three parameters available:

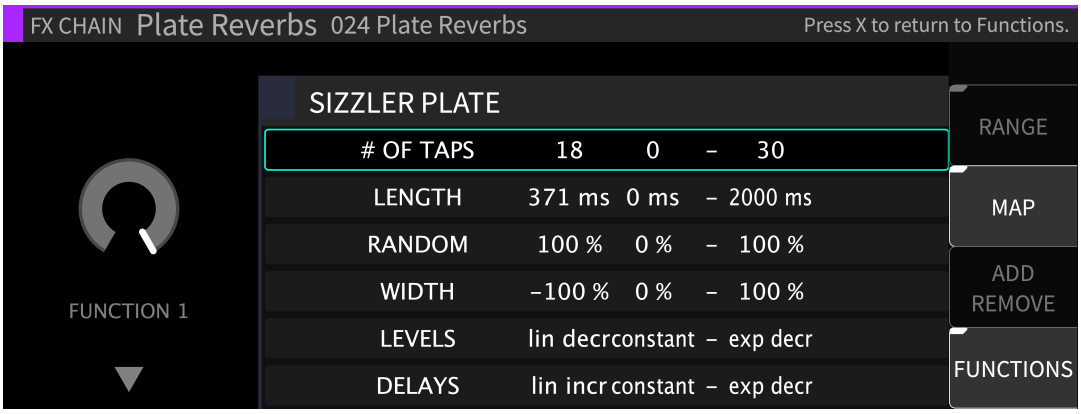
1. Range: Press the latching *Range* soft key to adjust the range of each parameter mapped to any given Function. Each parameter that belongs to a Function has a defined range; it can be as large as the true Minimum/Maximum range of the parameter, or anywhere in-between. The main Function knobs then cause each parameter change to be scaled according to its defined range.



You can use the *Min/Max* soft key to toggle between adjusting the minimum or maximum value of a parameter's range, as indicated by the white and green end cap markers. These Min/Max values only apply to parameters mapped to a Function knob, not the Function knob itself. To set a min/max value for a Function knob, use the *Custom Map* feature in the External Control settings.

As detailed earlier in the *Custom Map* settings, you can change the direction of a parameter's value by inverting the min/max end cap markers. This can be useful for having one Function knob control multiple parameters that move in different directions.

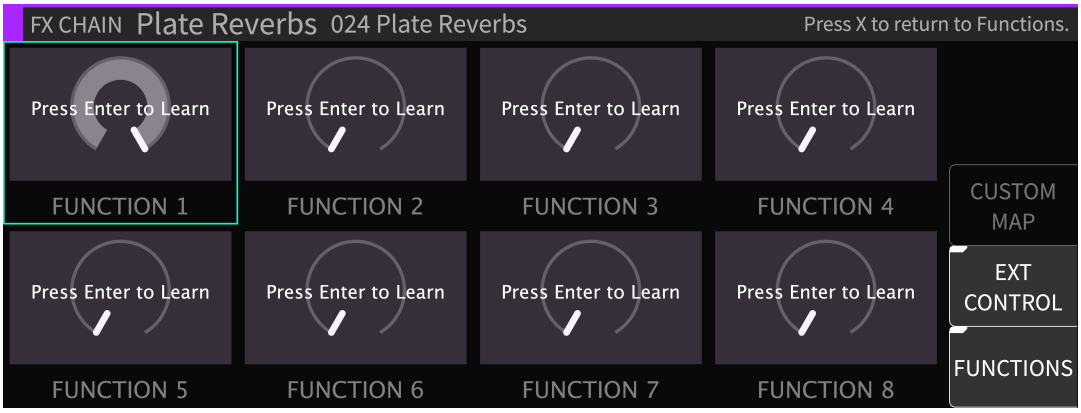
2. Map: Press the *Map* soft key to map specific parameters to the selected Function knob. The *Map* view displays a condensed view of all the parameters in the FX Chain, allowing their selection for assigning to Functions. The *Add/Remove* soft key assigns/unassigns the parameter of the selected Function.



- Highlight the Function knob on the left side of the display.
- Use the cursor up/down keys to select which of the eight Functions to work with.

- A list of available parameters will appear; use the cursor up/down keys to highlight the Function you wish to map.
- Press the *Add/Remove* or *Enter* key; the highlighted parameter is now mapped to the Function knob you had selected, and that Function's number (e.g., F1) now appears on the left side of that parameter's entry in the list.
- If a Function is already mapped to a parameter you have highlighted, pressing the *Add/Remove* soft key will remove the mapping.

3. Learn: Press the *Learn* soft key to map external controllers to any given Function knob.



- Use the cursor keys to highlight one of the eight Functions.
- Press the *Enter* key to learn a command.
- Move a control on your MIDI controller; that control will now be mapped to the highlighted Function.

The Custom Map dialog can also be accessed through this interface and used the same way as for individual parameters.

External Control

Many algorithms will feature an external control icon that looks like an expression pedal. While any controller can be mapped to any parameter on the H9000, the external controls are specific parameters that are intended to be mapped to a controller by the algorithm's creator. See the Algorithm Library for some examples of this.

To map an external control:

- Use the cursor keys to highlight the Configure button and press the *Enter* key.
- Follow the prompts to map your controller as described using the Custom Map dialog.
- Your controller will now be mapped to the designated parameter assigned by that algorithm's creator.

4.6. Meters

The H9000 offers an 8-channel set of dedicated three-segment hardware LED meters, positioned above the front panel color display. These LED meters are always active.

The first segment of the hardware LED meter represents signal presence, at any dB level. The second segment, if lit, means the signal is reaching -6 dB, which represents nominal headroom and nominal S/N ratio. The third segment indicates a signal overload, or too much signal presence.

For a more detailed metering view, press the front panel *Meters* button to switch the color display to the *Meters* page. The *Meters* page displays eight channels of metering at a time, and allows you to assign which signal sources are assigned to the display.

4.6.1. Assigning I/O sources to Meters

- On the main metering page, use the cursor buttons or the wheel to navigate to the metering slot to which you wish to assign an audio input or output.
- Press the *Enter* key, or the latching *Assign* soft key, to make an assignment.
- You will be presented with a list of every I/O channel; use the cursor up/down buttons or wheel to navigate to the physical input you wish to assign.
- Press the *Enter* button to accept your choice The selected input/output is now assigned to the highlighted metering channel, and that channel will now show the assignment above the channel number.

More Meters

Press the latching *More Meters* soft key; the display changes to show metering data for the currently loaded available FX chains. The *More Meters* screen also has soft keys for the following functions:

- Levels: Press this latching soft key to switch to the *Levels* screen.
  - You are presented with input and output level faders for the currently selected FX chains.
  - Use the cursor buttons to navigate to the input or output fader you wish to adjust, then use the wheel to adjust the highlighted fader.
  - On the levels screen, press the latching *Group* soft key to group together level faders. When the latching soft key is illuminated, adjusting the level of either group's fader will adjust the level of all faders in each grouping.
- Clear Clips: Press this soft key to clear any overload clips that have occurred.
- I/O Meters: Press the latching *I/O Meters* soft key to display the I/O meters screen; you will be presented with metering views for the H9000's banks of physical inputs and outputs, regardless of what custom assignments you have made in the main metering screen.

The I/O meters screen shows which bank of I/O metering you are on in the upper left corner, and which of the 8 pages you are on in the upper-right corner.

On a stock H9000 with no expansion cards installed, the following views are available:

- Digital Audio In 1-8
- Analog In 1-8
- USB In 1-8
- USB In 9-16
- Digital Audio Out 1-8
- Analog Out 1-8
- USB Out 1-8
- USB Out 9-16

Use the cursor up/down buttons or wheel to navigate among the eight pages of metering views.

Note

While in the I/O Meters page, you can use the navigation keys to select a meter. If a physical input or output is routed to or from an FX chain, highlight the meter to see which FX chain it is routed to or from. This is indicated by a purple square under the meter and the corresponding label bar at the bottom of the page.

4.7. Saving and Importing Data

The H9000 allows you to save various types of data, both to its internal memory or a USB drive connected to any of the USB ports.

Once you have saved this data to an attached USB drive, you can import it to a different H9000, allowing you to share your work among machines.

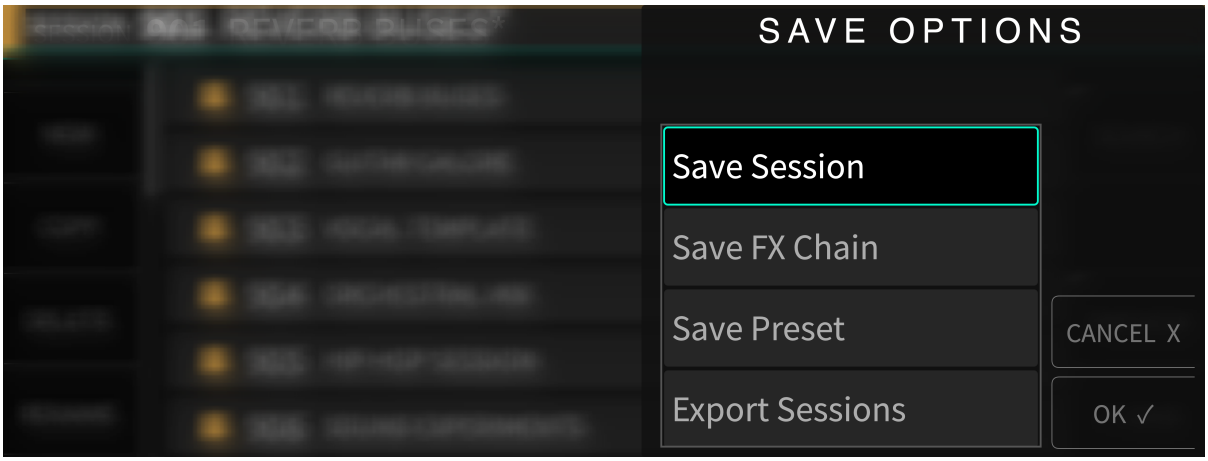
Note

It is recommended to use the included FAT32-formatted USB drive. Please contact support@eventide.com if you need assistance formatting another USB drive.

4.7.1. Saving

Press the front panel *Save/Import* button. The *Save* screen will appear and you will be presented with the following options:

- Save Session
- Save FX Chain
- Save Preset
- Export Sessions
- Export Presets
- Export FX Chains
- Export Algorithms



The data to be saved is contextually based on the currently selected item, as follows:

- *Save Session* will always save the currently loaded session.
- For *FX Chains* and *Presets*, the current FX chain and algorithm container selected on the screen will be the items saved for each save option.

At any time, you can refer to the cursor highlight or the title bar in the *FX Chains* or *Algorithms* view in order to see the currently selected item.

- Navigate to one of the choices above and press the *Enter* key to access the naming screen.
- Use the front panel keypad to type a custom name for your saved data, then press the *OK* soft key. The data will be saved to the H9000's internal storage.
- If a USB drive is attached, you will then be presented with the option to also save the data to the drive as well.
- Press the *YES* soft key to save your data to the attached USB drive.
- To backup your H9000, refer to Section A.4: USB Dump Data.

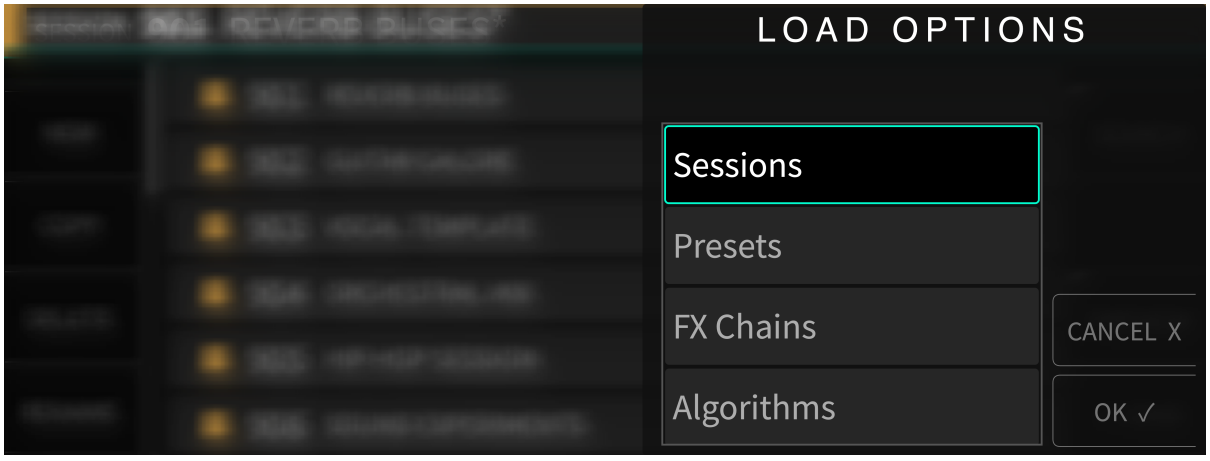
4.7.2. Exporting

- If you have a USB drive plugged in, you will also see the “Export” options. The export options will *NOT* appear in the list if no drive is attached.
- Whereas “Saving” refers to storing something onto the H9000’s internal memory, “Exporting” transfers it from the H9000 to the USB drive.
- When saving with a USB drive plugged in, you’ll also be given the option to export after the save is complete.

4.7.3. Importing

Once you have saved H9000 data to a USB drive, that data can be imported to any H9000 unit by connecting the drive. Importing works as follows:

- Attach a USB drive containing H9000 data to a USB port of your H9000.
- Long-press the front-panel *Save/Import* button; the *Load Options* screen will appear.
- Use the cursor up/down buttons or wheel to navigate to the specific type of data you wish to load, then press the *Enter* key.
- You will have the option to load Sessions, Presets, FX Chains, and Algorithms.
- Use the cursor up/down keys or wheel to select the data you wish to load, then press the *SELECT* key; the selected data will be loaded into memory.
- If you wish to load all of the data in the list, press the *SELECT ALL* soft key instead, or press the *CANCEL* soft key to cancel out of the load operation.



Note

Any algorithms you load would be sourced from a development environment, such as Eventide's VSig software editor. Once you have registered your H9000, the VSig software will be available for download at: https://etide.io/vsig3

4.8. Scenes

Long press the front panel *Tap/Scenes* button to bring up the SCENES screen.

Scenes are used to control sequenced program changes in the H9000. This mode can be useful when you are using the H9000 in a live environment such as a concert or theatrical production, and wish to "step through" different pre-configured states in an order you have set up in advance.

For example, the main vocalist in a concert may need different reverb parameters at different times during the show; scenes provide a quick, simple way to achieve this:

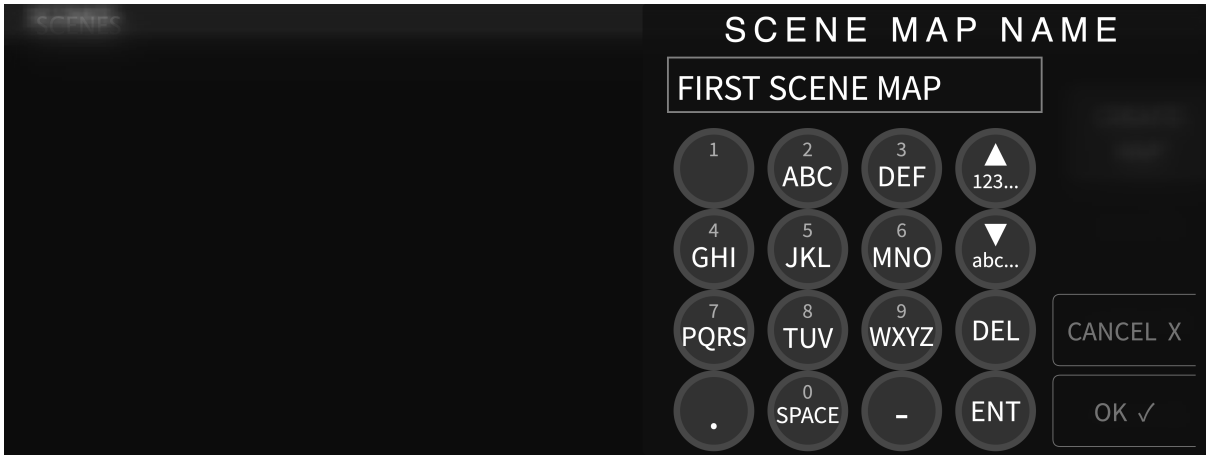
- You can create multiple "Scene Maps" and select one at any given time
- A Scene Map is a collection of Scenes, along with the Trigger Sources
- Each Scene Map can contain a total of 128 Scenes
- A Scene itself can be thought of as a particular *state* of the FX chains, running under a current session
- The "Scene" denotes a set of actions to be triggered that put the H9000 into that particular state, either an:
  - FX chain or algorithm preset to be loaded into a destination container
  - A mute/bypass/enable action to be applied to a destination container

4.8.1. Working with Scenes

The right side of the screen displays the 128 available scenes, while the left side of the screen displays a list of Scenes Maps you have created. When entering SCENES mode, the following soft keys are available for Scene Maps:

Create Map

Press this soft key to create a new Scene Map. You will be prompted to name the Scene Map, at which point it will be added to the list of created Scene Maps on the left.



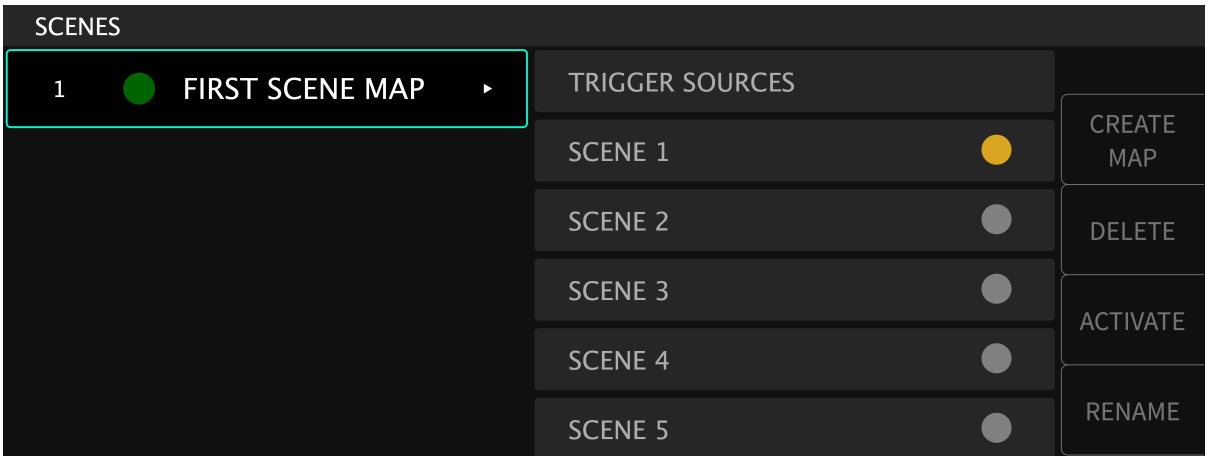
Delete

Press this soft key to delete the highlighted scene.

Activate

Press this soft key to activate the highlighted Scene Map. Once a Scene Map is activated, a green circle will appear to the left of its text, indicating that it will now respond to MIDI messages.

To de-activate a Scene Map, press the *Activate* soft key while highlighting the currently active Scene Map. The circle to the right of its text will turn red, indicating that it is not active and won't respond MIDI messages. Alternatively, you can activate a different Scene Map.



Note

Only one Scene Map can be active at any given time. When a Scene map is not active it will ignore all MIDI messages.

Rename

Press this soft key to rename the highlighted Scene Map.

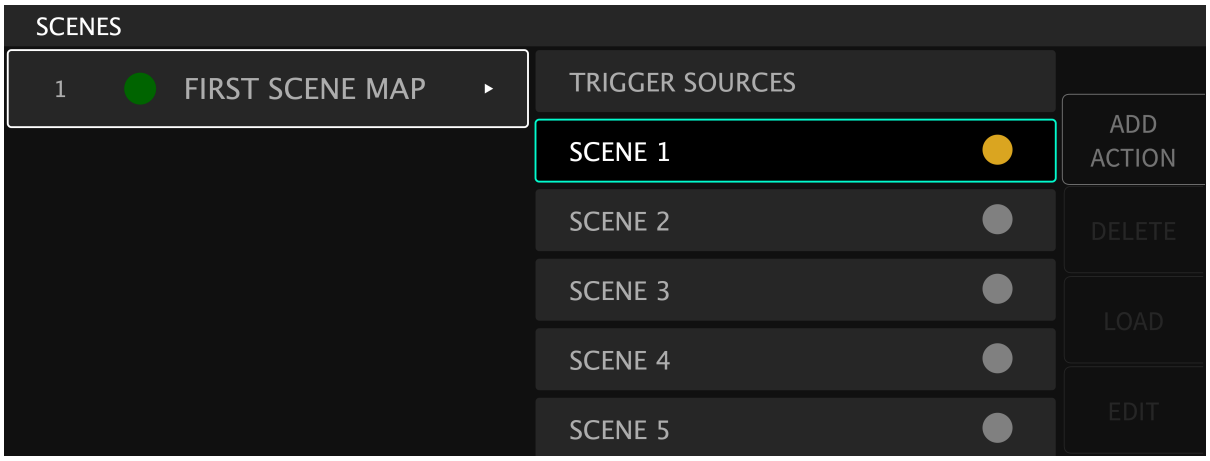
4.8.2. Scene Maps

Once a Map is activated, the following soft keys are available for its contained scenes:

Add Action

Highlight a scene on the right-hand side, and press the *Add Action* soft key. You will be presented with the choices of:

- Scene Capture
- Manual



Scene Capture

This is a way to take a “Snapshot” of the current settings of the Session for a Scene.

Manual

This allows you to create a Scene from a blank template.

Delete

Press this soft key to delete a highlighted scene.

Load

Press this soft key to activate the highlighted scene. The circle to the right of the scene text will turn GREEN, indicating that this scene is now active.

Rename

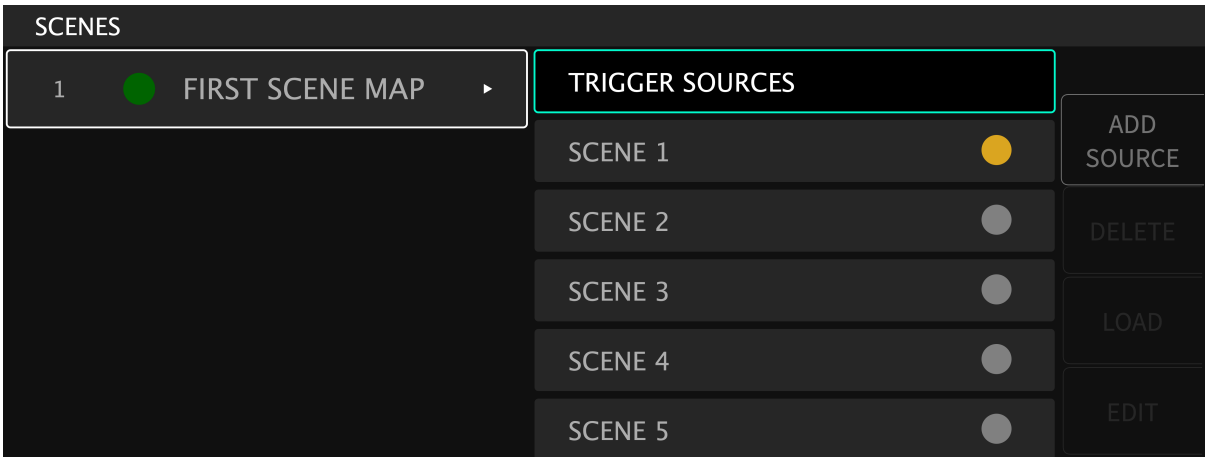
Press this soft key to give your Scene a custom name.

Edit

Press this soft key to edit the highlighted scene. You will have the options to edit with either the Scene Capture or Manual functions.

4.8.3. Trigger Sources

At the top of the Scene Map list on the right-hand side, you'll see the "Trigger Sources" tab. Highlight this tab, then press the *Add Source* soft key. You will be prompted to select a source mode, message type, and MIDI channel.



- Select Source Mode: You are presented with the source mode options of:
  - Load by number
  - Increment
  - Increment and load
  - Decrement
  - Decrement and load
  - Load current
- Select Message Type: Use the cursor up/down keys or wheel to select an option from the above list, then press the *Enter* key. You will then be presented with the option to trigger the action using:
  - Program change
  - Note on
  - CC (MIDI Continuous Controller)
  - Aux switch
- Select MIDI channel: Use the cursor up/down keys or wheel to select an option from the above list, then press the *Enter* key. You may then choose from which MIDI channel to receive messages.

4.8.3.1. Trigger Source Options

A “Trigger Source” allows assignment of actions, such as loading a Scene by number, or incrementing/decrementing. You can control Scene changes from the front panel, using Emote, or an external control source such as a MIDI controller or Aux Switch.

Scenes can be triggered in the following ways:

Load By Number

Load By Number works like a traditional program change message; it will load whatever Scene matches the number of the program change message you send, between 0 and 127. Although it usually makes sense to use actual MIDI Program Changes, you can also use Note On or Continuous Controller triggers.

Increment/Decrement

Increment/Decrement switches to the next or previous scene, but keeps it in a pending state. When a scene is in a pending state, the circle to the right of its text will be yellow.

Load Current

Loads a scene from its pending state after being selected with the above mentioned “Increment/Decrement” function. When a scene is loaded from a pending state, the circle to the right of the text will become green, indicating the scene has been loaded.

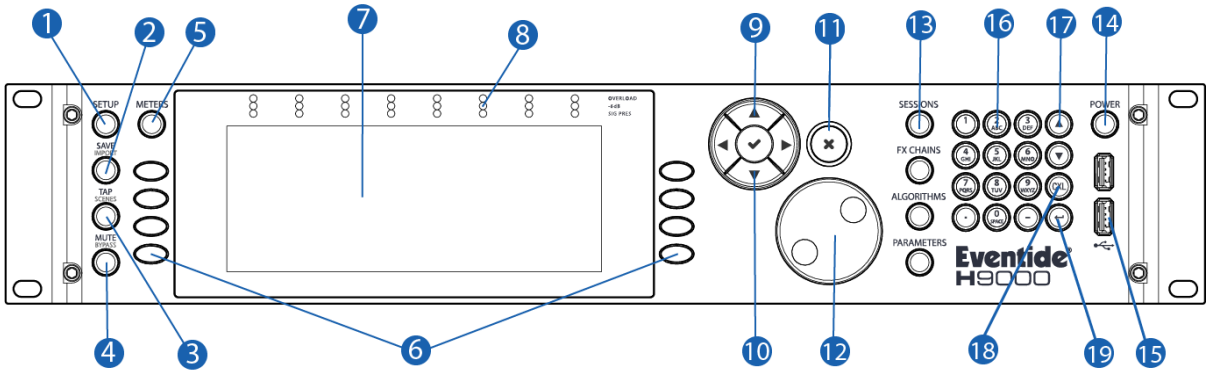
Increment & Load/Decrement & Load

Switches to the next or previous scene and then loads it immediately. If your scenes are sequenced in the order to be used, it’s probably easier to use the Increment & Load/Decrement & Load options. However, if you need to jump around or skip Scenes, you can use Increment/Decrement with the Load Current option.

This page was intentionally left 99.88% blank.

5. HARDWARE

5.1. H9000 Front Panel



The H9000 front panel contains the following controls:

1. Setup

   Press the *Setup* button to enter the setup menu, where you can configure preferences for your device.

2. Save/Import

   Press the *Save/Import* button to save the different types of presets to internal memory, as well as optionally saving to a FAT32-formatted USB drive. Long-press the *Save/Import* button to import saved presets into the H9000. See the section “Saving and Importing” for more details.

3. Tap/Scenes

   Press the *Tap* button multiple times to set a BPM tempo (BeatsPerMinute). This tempo setting will be used as a sync reference for time-based effects such as digital delay.

   The *Tap* button’s inset LED illuminates in time with the tempo you have set, giving a visual indication of the tempo. Note that if you’re not a fan of constant flashing, this can be disabled in the *Setup* menu.

Long-press the *Tap/Scenes* button to bring up the SCENES screen. See section 4.8: Scenes for more details.

4. Mute/Bypass

   Press the *Mute/Bypass* button in order to mute the audio output of the H9000's FX Chains. Long-press the *Mute/Bypass* button in order to bypass all processing while still preserving the flow of audio through the I/O connectors and FX chains.

   When either Global Mute or Global Bypass is activated, an indication will illuminate in the upper right hand corner of the display until deactivated.

   This has no effect on Direct I/O (Section 4.3.3).

5. Meters

   Press the *Meters* button to switch the display to a full meter bridge.

6. Soft Keys

   These eight unlabeled buttons (four on either side of the display) allow you to quickly navigate the H9000's front panel user interface. The soft keys correspond to different functions and are always context sensitive to the currently selected screen.

7. Color Display

   The H9000's color display allows you to both view and control all aspects of the product while working with its front panel controls. The color display on the front panel is *not* a touchscreen.

8. 8-Channel Meters

   These three-segment hardware meters display audio level information for the H9000 and are active at all times.

9. Cursor Diamond

   Use the four keys of the cursor diamond to navigate up, down, left and right.

10. Enter (√)

    Press the *Enter* (check mark) button to confirm or complete an action that you just set up.

11. Cancel (X)

    Press the *Cancel* (X) button to cancel out of an action.

12. Jog Wheel

    The large, weighted jog wheel is used to both navigate around the display, as well as to adjust a currently highlighted setting. For example, if adjusting a room size parameter in a reverb algorithm, you can spin the wheel to adjust the specific parameter value.

13. Mode Buttons

    The H9000 has four dedicated Mode buttons (**SESSIONS**, **FX CHAINS**, **ALGORITHMS**, and **PARAMETERS**) that let you switch to any of the editing modes with a single touch. The selected button illuminates to indicate the current mode.

14. Power

    Press the latching power button to turn the H9000 on or off.

15. USB-A Ports

    The front panel USB ports provide three separate functions:

    - Connect a FAT32-formatted USB thumb drive to either of the front/rear panel USB ports to save any H9000 presets you have created or perform an offline firmware update.
    - Attach the included USB Wi-Fi adapter to connect the H9000 to a local area network, or to utilize the Wi-Fi access point.
    - Connect a class-compliant USB MIDI controller for real-time adjustment of many H9000 functions.

16. Number Pad

    Use the buttons on the number pad to enter a custom name when presented with a naming screen. The number buttons operate “vintage mobile phone” style, where each button cycles through multiple letters and also offers special characters.

17. Up/Down Arrows

    The up/down arrows act as modifier keys for the number pad when naming an object, allowing you to switch between upper and lower case characters as well as to enter different symbols. You can also use the arrow keys to fine-tune parameter values, as well as scroll through list items.

18. CXL

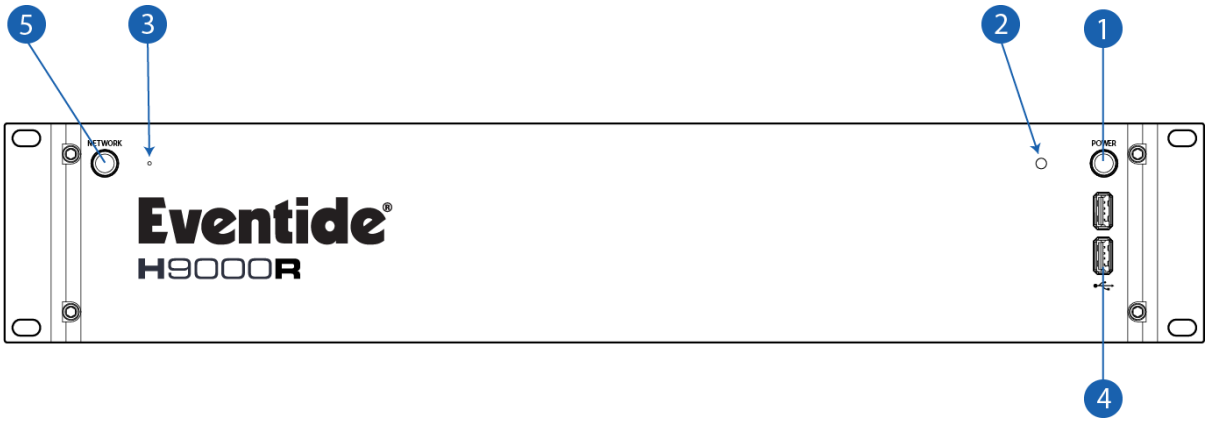
    Press the CXL button to delete a character you have typed when creating a custom name. Long press the CXL button to clear the entire text field.

19. Return

    Press the return arrow to confirm the name you entered in the naming field.

5.2. H9000R Front Panel

The H9000R front panel has the following controls:



1. Power

   Press the latching power button to turn the H9000R on or off.

2. Power LED

   This LED illuminates when the H9000R is powered on.

3. Reset Button

   Use this pinhole key in the event you need to perform a failsafe boot (as described in Section A: Startup Sequences).

4. USB-A Ports

   The front panel USB ports provide three separate functions:

   - Connect a FAT32-formatted USB thumb drive to either of the front/rear panel USB ports to save any H9000R presets you have created or perform an offline firmware update.
   - Attach the included USB Wi-Fi adapter to connect the H9000R to a local area network, or to utilize the Wi-Fi access point.
   - Connect a class-compliant USB MIDI controller for real-time adjustment of many H9000R functions.

5. Network Button

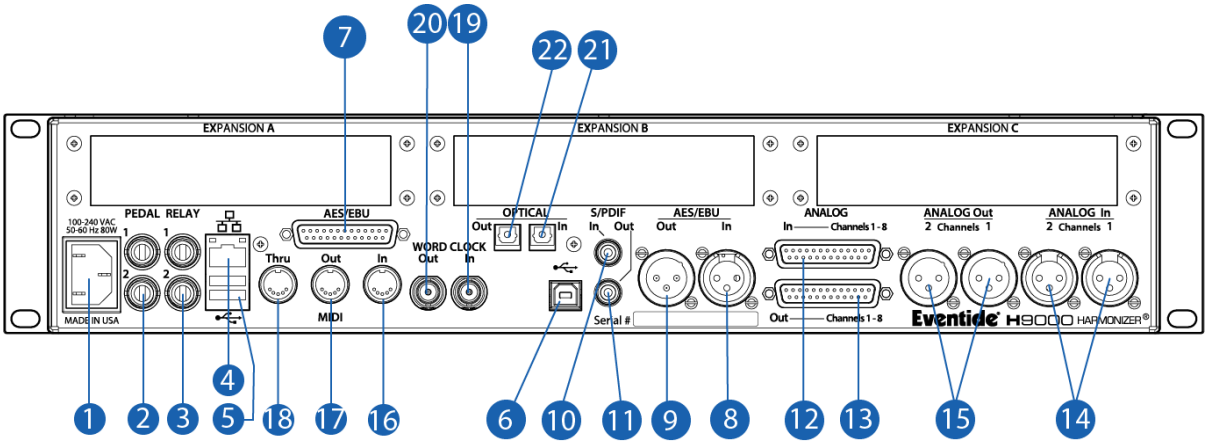
   To the left of the pinhole is the Network Button. The LED in the center of this button will flash while the H9000R is starting up. This button is used to unlock the Network Settings (Section 3.1.2.3) in Emote, and to perform various Startup Sequences (Section A).

Note

After pressing the power button on your H9000R, the Network button's LED will flash for about two minutes, then be dim for a few seconds, then brighten. Your H9000R will appear in Emote once the Network LED has brightened.

5.3. Rear Panel

The H9000 and H9000R have identical rear panels. They offer a wide variety of audio and data connections, allowing them to be used in many different environments and workflows.



1. Power Connector

   Connect the H9000 to your power source using a standard IEC power cable. The H9000's power supply is designed to work from 100VAC to 240VAC, 50-60Hz, without adjustment. Your unit will ship with a power cable designed to work in your territory.

2. Pedal (x2)

   The H9000 has 2 ¼" TRS inputs that allow for a variety of external control sources including expression pedals, foot switches, control voltages, or a combination.

3. Relay (x2)

   The dual ¼" relay connectors allow the H9000 actions to close external circuits. Currently, the relay connectors on the H9000 are not functional. They will become useable in a future revision.

4. Ethernet Connector

   This standard RJ-45 connector can connect the H9000 directly to a computer. It can also connect to a local area network via a router, or a switch such as the Dante expansion card (Section 7.2). Once connected, you can view and control the H9000's operations (Section 3.3), and update your H9000's firmware using Emote (Section 6.1).

5. USB A Connector

The rear panel USB A ports provide the same functions as the front panel USB A ports.

6. USB B Connector

Connect the H9000 to your Mac or PC, allowing you to use it as a 16-input, 16-output computer audio interface (Section 2.2).

> **Note**
>
> Due to its design, USB cable has a normal maximum length of about 5 meters (16.4 feet), but you can get longer distances by using an Active USB Extension/Repeater Cable. Exceeding this maximum length without an Active USB Extension/Repeater Cable may cause issues. If you do use a USB extension, be sure it is a USB 2.0 or 3.0 extension, and not a USB 1.0 extension.

7. AES/EBU Connector

This female DB-25 connector allows you to feed the H9000 with eight channels of AES-EBU digital audio, and also allows the unit to output eight channels. TASCAM Standard pinout: https://etide.io/DB-25-Pinout

> **Note**
>
> When connecting the H9000 to another device via the AES-EBU connector, you must use the correct AES-EBU DB-25 cable. If the other piece of equipment uses Yamaha Standard pinout, you must use an AES-EBU TASCAM to Yamaha crossover cable. Please refer to your equipment's documentation to determine what pinout standard it uses.

8. AES/EBU 2-channel Input

This female XLR input accepts a standard 2-channel AES/EBU digital audio signal.

9. AES/EBU 2-channel Output

   This male XLR connector outputs 2 channels of audio in the AES/EBU digital audio format at the H9000's current sample rate.

10. S/PDIF Input

    This RCA connector input accepts a standard 2-channel S/PDIF digital audio signal.

11. S/PDIF Output

    This RCA connector outputs 2 channels of audio in the S/PDIF digital audio format at the H9000's current sample rate.

12. Analog Input 8-channel

    This female DB-25 connector feeds the H9000 with 8 channels of balanced analog audio input. TASCAM Standard pinout.

13. Analog Output 8-channel

    This female DB-25 connector outputs 8 channels of balanced analog audio. TASCAM Standard pinout: https://etide.io/DB-25-Pinout

14. Analog Input 2-Channel

    These dual balanced female XLR connectors accept two channels of balanced analog audio.

15. Analog Output 2-Channel

    These dual balanced male XLR connectors output two channels of balanced analog audio.

16. MIDI Input

    The MIDI input allows you to control various H9000 parameters with incoming MIDI messages, using MIDI controller hardware or MIDI software of your choosing. The MIDI input also allows you to synchronize the H9000's time-based effects to an incoming MIDI clock signal. See MIDI Functionality: (Section C)

17. MIDI Output

    The MIDI Out connector outputs MIDI continuous controller messages, allowing for MIDI feedback capabilities.

18. MIDI THRU

    The MIDI THRU connector accepts incoming MIDI signals from the MIDI DIN input and outputs a mirrored duplicate of those signals. This allows you to daisy chain multiple MIDI devices.

19. Word Clock Input

    This BNC connector allows the H9000 to sync its digital system clock to an incoming word clock signal. It is compatible with word clock signals at 44.1kHz, 48kHz, 88.2kHz, and 96kHz.

20. Word Clock Output

    This BNC connector outputs the H9000's digital clock signal at the unit's currently configured sample rate.

21. Optical Input

    The optical input accepts 8 channels of ADAT format digital audio, or 2 channels of S/PDIF format digital audio.

22. Optical Output

    The optical output sends 8 channels of ADAT format digital audio, or 2 channels of S/PDIF format digital audio.

> **Note**
>
> The H9000's optical connectors only support sample rates of 44.1kHz and 48kHz; they are not operational when the H9000 is set to run at higher sample rates of 88.2kHz or 96kHz.

23. Expansion A,B,C

    These 3 slots allow for optional expansion cards detailed in Section 7: Expansion Cards.

This page was intentionally left 99.88% blank.

6. FIRMWARE

6.1. Updating Firmware Using Emote

> **Note**
>
> The firmware update process is the same for H9000 and H9000R units.

To update via the internet, your H9000 must be connected to Emote. This can be via a networked connection using ethernet or Wi-Fi, or a direct ethernet connection to your computer. Do not interrupt the network connection to your H9000 or close Emote during the update process.

Once a connection to Emote has been established:

- Go to the Devices View (the button labeled 9K in the menu bar) and locate the H9000 Firmware window at the bottom left corner of the page.
- Click on “Updates” and you will be presented with a list of available firmware updates to choose from.
- Select a firmware version and click the “Install” button; a pop-up message will confirm that you are updating your H9000’s firmware. The first progress bar will indicate that the firmware update file is being downloaded and sent to the H9000. The second progress bar will indicate that the firmware is being installed on the H9000. This will also be reflected on the front panel.
- Once the firmware update has been successfully installed, you will be prompted to restart your H9000, which will complete the update process.
- Please restart Emote after the H9000 has rebooted before attempting to connect to an H9000 that has just been updated.

In the event that you need to downgrade your H9000’s firmware, please contact support@eventide.com and we will provide you with the correct firmware version to install via the offline USB updater.

⚠ Warning

Please be sure not to interrupt power to your unit during the update process; doing so could damage its internal firmware, and result in putting the unit into an unbootable state. In the event this does occur, refer to Section A.2 Recovery Mode to reinstall the firmware on your H9000. It is advised to save your current session before updating the H9000's firmware.

6.2. Updating Firmware Without an Internet Connection

Refer to Section A.2 Recovery Mode for information on updating firmware without an internet connection.

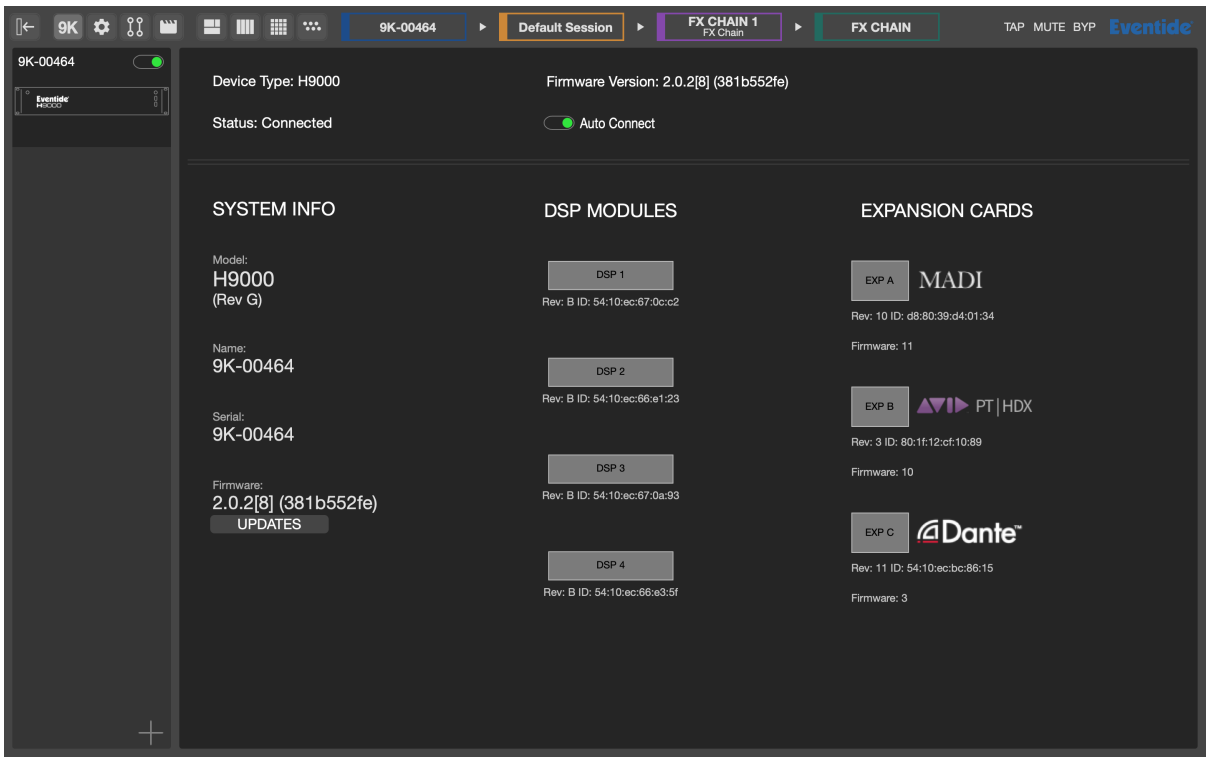
6.3. Updating Legacy Firmware

Please refer to https://etide.io/H9000-Legacy-Update if you are updating your H9000 from firmware version 1.3 or below.

7. EXPANSION CARDS

The H9000 offers additional I/O via three expansion card slots with up to 32 channels of bidirectional audio each. Installation instructions can be found here: https://etide.io/H9000-EXP-Guide

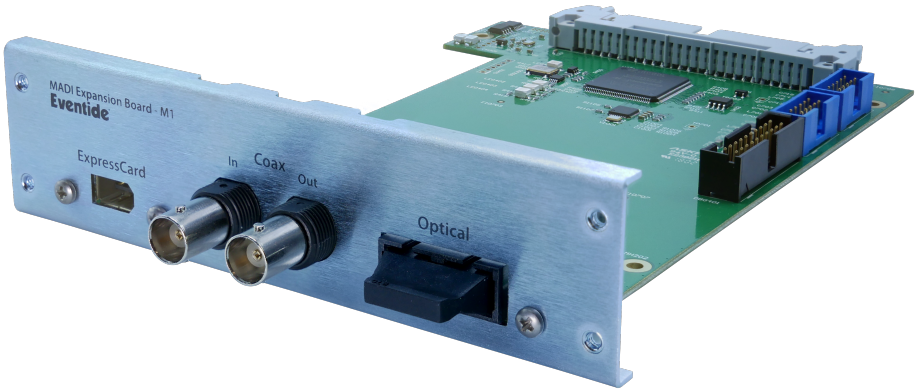
Once installed, your expansion card(s) will be listed on the front panel's Setup page and in Emote's Devices View, and will appear as I/O channels within Emote and on the front panel.



Note

There is no Eventide registration process for H9000 expansion cards.

7.1. MADI



- Supports full AES5 compatibility.
- Sample Rate Clock Sources: H9000 or the selected source.
- 32 In / 32 Out operation from 44.1kHz to 96kHz.
- Optical, Coax, or ExpressCard (RME proprietary) connections.

Note

MADI coax requires 75 ohm cables, but is point-to-point only and does not allow daisy-chaining. Thus, it does not require any external termination. Termination is done internally on both transmitter and receiver ends. T-connectors should never be used.

7.1.1. MADI Setup

These options are available in the Setup page for the MADI Expansion Card:

Source Select

Selects which source the MADI card receives audio from: Optical, Coax, or ExpressCard. The MADI card sends audio to Optical, Coax, and ExpressCard simultaneously.

Tx Format

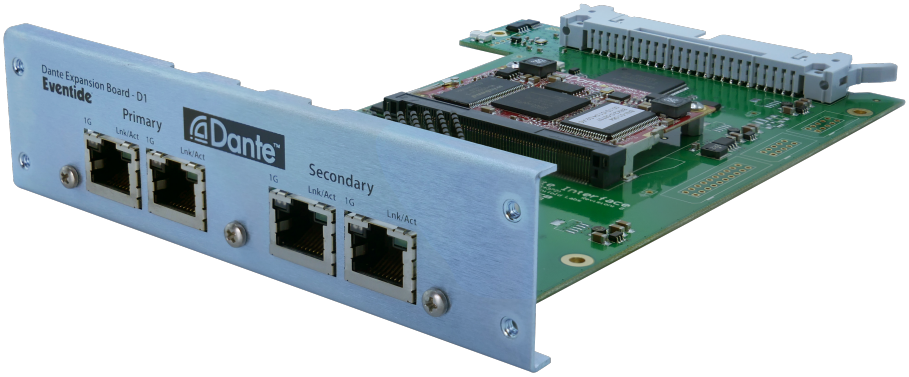
Auto/56, Auto/64, 56 ch, 64 ch.

Lock Status

Indicates whether your selected port is connected or not. If GREEN, your MADI Expansion card is connected to your MADI interface. If RED, your MADI Expansion card is not connected or being detected by your MADI interface. Check your connections.

> **Note**
>
> The MADI expansion card only uses channels 1-32 and cannot be daisy chained after another MADI device.

7.2. Dante



- Audinate Brooklyn II Dante Module based design
- 32 In / 32 Out operation from 44.1kHz to 96kHz
- Sample Rate Clock Sources: Brooklyn II Module, Dante network, H9000
- AES67 Support
- Either 2 Primary and 2 Redundant Networks Connections or 4 Primary Network Connections

- Integrates with Dante-enabled devices: https://etide.io/dante-enabled

7.2.1. Dante Setup

7.2.1.1. Clock Sync

From a clocking point of view, it is best to think of the Eventide H9000 and its Dante expansion card as two separate devices.

H9000

Once a Dante card is installed, “Exp A/B/C (Dante)” will appear as a clock source option in Emote (Section 3.3.8), and on the front panel (Section 2.1.4). This option syncs the H9000’s clock to the Dante card’s clock.

Dante

The Dante card’s clock is set in Audinate’s Dante Controller, which is available here: https://etide.io/latest-dante Dante Controller’s Clock Status tab displays the current clock source of the Dante card. This clock source can be set with the “Enable Sync To External” checkbox.

To sync the H9000 to Dante, set the H9000’s clock source to “Exp A/B/C (Dante)”, and uncheck “Enable Sync To External” for the Dante card in Dante Controller.

To sync Dante to the H9000, set the H9000’s clock source to “Int {sample rate}”, and check “Enable Sync To External” for the Dante card in Dante Controller.

7.2.1.2. Sample Rate Match

The Dante card’s sample rate must be set manually under the Device Config tab in Dante Controller’s Device View.

If the H9000’s clock syncs to Dante, then the H9000’s clock will automatically follow Dante’s sample rate.

If Dante's clock syncs to the H9000, then the sample rate in Dante Controller's Device View must be manually configured after changing the H9000's sample rate.

7.2.2. Documentation

https://etide.io/dante-tech-docs

> **Note**
>
> The Dante expansion card is not a network router, but it is a network switch. If Emote is on the same local area network as Dante, and the Dante expansion card is configured in Switched mode, then an ethernet cable can be plugged in directly between the H9000's ethernet port and any of the Dante card's primary ports.

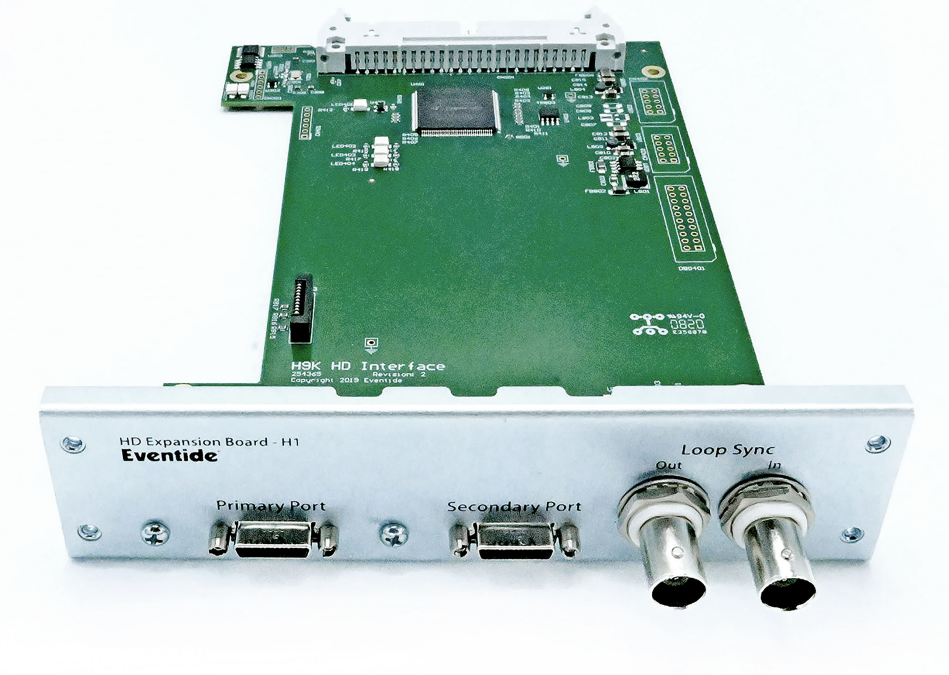
7.2.3. Firmware updates

The H9000 Dante expansion card firmware update is available here: https://etide.io/dante-exp-firmware

To update your H9000 Dante expansion card firmware, first download Audinate's Dante Controller, available here: https://etide.io/latest-dante

The Updater is available in Dante Controller's "view" menu. Import "h9k-dante-io-board-19.1.31.1.dnt" into the Updater by following Audinate's guide here: https://etide.io/dante-firmware-import

7.3. Pro Tools | HD



- 2 DigiLink Mini ports
- DigiLink Connections feature a Primary and a Secondary Port which is switchable between 32- and 16-channel I/O
- LoopSync In and Out connectors
- LoopSync Connections allow you to use Pro Tools to assign the clock source as either the H9000 or another HD device
- 32 In / 32 Out operation from 44.1kHz to 96kHz
- Includes 12’ DigiLink Mini cable

7.3.1. Pro Tools | HD Setup

These options are available in the Setup page for the Pro Tools | HD Expansion Card:

Sample Rate

Displays the sample rate being output by Pro Tools. This will change depending on the Sample Rate of the Pro Tools Session.

Clock Source

Displays either “Internal” or “LoopSync”. If the H9000 is the *Clock Source* in Pro Tools, then this will display “Internal”. If another HD device is the *Clock Source* in Pro Tools, then this will display “LoopSync” if a LoopSync connection has been established. These settings will also be reflected on H9000’s *Clock* menu page.

> **Note**
>
> Sample Rate and Clock Source are both read only parameters on the H9000’s front panel and in Emote. These parameters can only be changed using the Pro Tools *Hardware Setup* page, or in your computer’s audio configuration page, such as the Audio MIDI setup page on macOS.

Interface Mode

Toggles between 32 Channel mode and 16 Channel Mode. In 32 Channel Mode, the expansion card will output 32 channels through the Primary DigiLink port. Use this setting if you are connecting directly to the HD Card to take advantage of all 32 channels. In 16 Channel Mode, the expansion card will output 16 channels through both the primary and secondary port. Use this setting if you are daisy chaining into another HD interface’s secondary port since the maximum channel count per DigiLink port is 32.

Connection Status

Indicates DigiLink port connection status. If GREEN, your HD Expansion card is connected to your HD System. If RED, your HD Expansion card is not connected or being detected by your HD System. Check your connections.

Loop Master Status

Indicates what source is being used as the “Loopmaster” in the LoopSync. If GREEN, your H9000 is serving as the “Loop Master” to the rest of your HD System. If RED, your H9000 is clocking to another source through the LoopSync of your HD System.

Clock Control

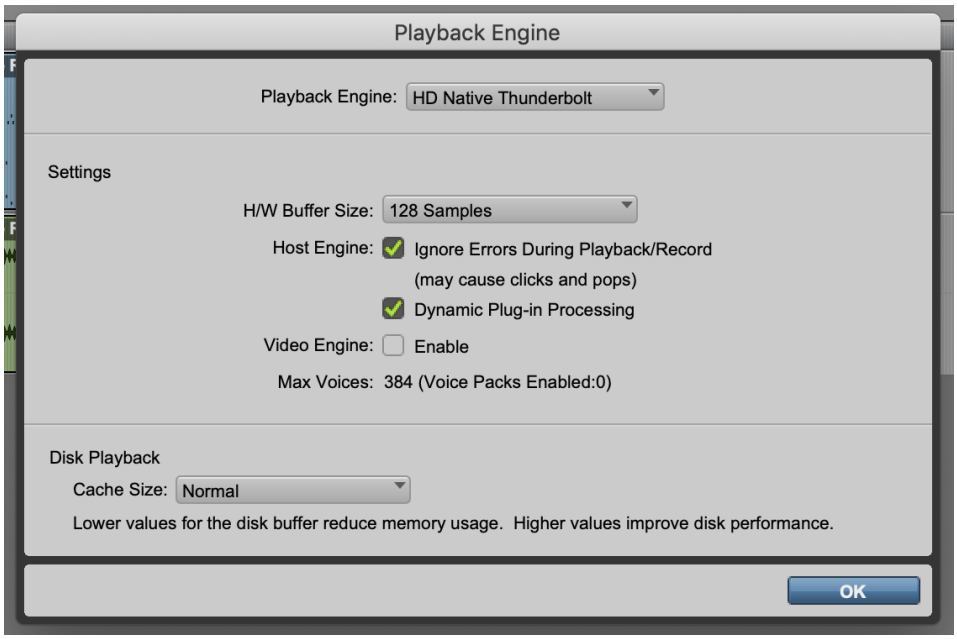
Allows Pro Tools permission to control your H9000’s internal clock via the HD expansion card. When using Pro Tools, toggle this on so that your H9000 can sync seamlessly with your HD System’s clocking. When not using Pro Tools, turn this off so that the H9000 will ignore any clock changes you make from Pro Tools.

Note

When Clock Control is enabled, you will not be able to change the H9000’s clock settings regardless of whether the HD expansion card is being used or not. Turn Clock Control off to change the H9000’s clock settings.

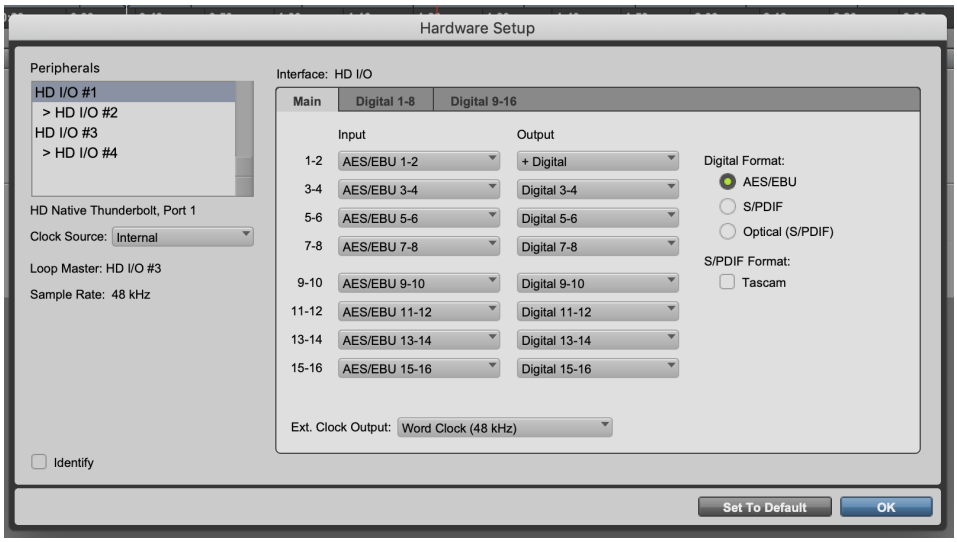
7.3.2. Integration

To use Pro Tools | HD I/O with the H9000, you must connect the expansion card to your HD system via the DigiLink ports. Once this is connected, You must select the HD device as your *Playback Engine* in Pro Tools. In the example below, we are using an HD Native Thunderbolt to integrate it into the HD environment.



Once you have the HD device selected as your playback engine, you can now route audio to and from the H9000 via the expansion card's DigiLink ports.

To use the H9000 as a H/W insert in Pro Tools, first you have to check your *Hardware Setup* page in Pro Tools to see which inserts the H9000 will correspond to.



On the left side of the *Hardware Setup* page, your HD devices will appear in the *Peripherals* window. In this case, we have our HD audio interface plugged into the Primary port, and the H9000 plugged in to the Secondary port of the HD Native. In the *Peripherals* window, the HD audio interface will show up as HD I/O #1 and #2, and the H9000 will show up as HD I/O #3 and #4.

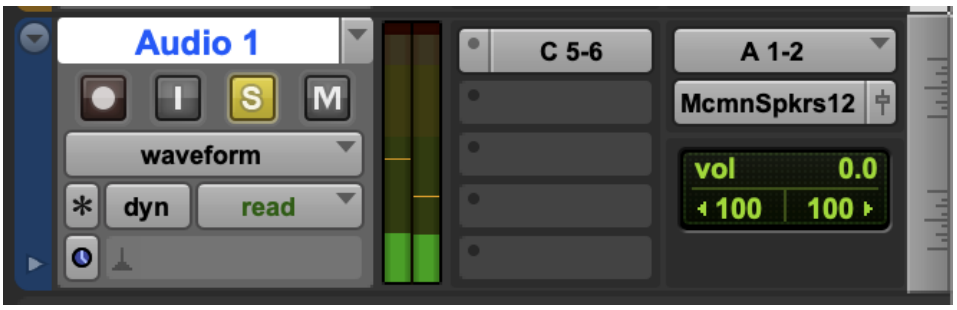
Highlighting a device in the *Peripherals* window will allow you to view its I/O settings. In Pro Tools, the hardware inserts for HD I/O #1 and #2 will appear as **A 1-16** and **B 1-16**. Since the H9000 is plugged into the Secondary port (HD I/O #3 and #4), its corresponding hardware inserts will be **C 1-16** and **D 1-16**. Note that the I/O of the HD expansion card will show up in the Pro Tools *Hardware Setup* page as "Analog 1-16" when these are, in fact, digitally routed to the H9000.

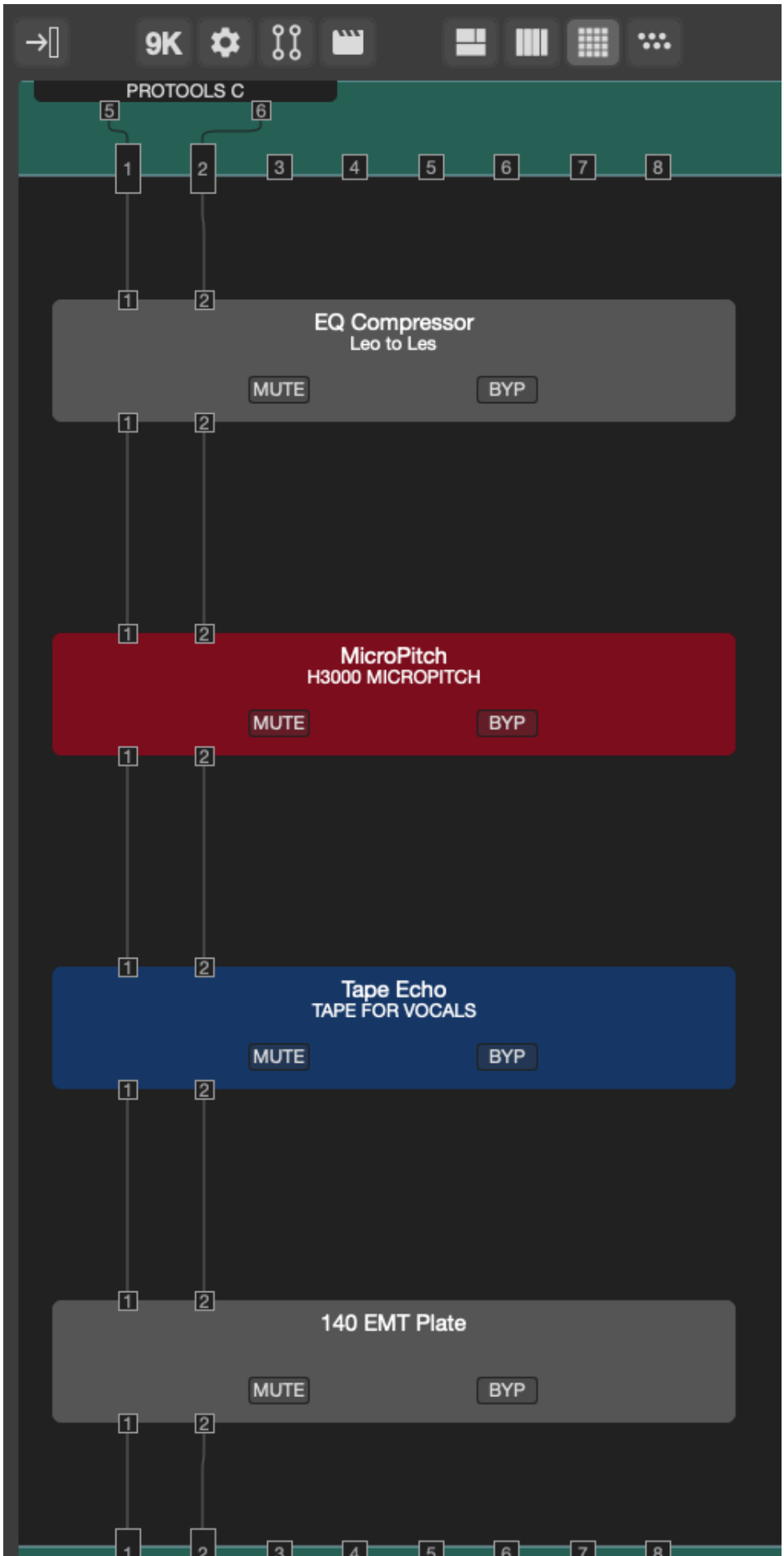
Note

In Pro Tools, 1st party HD interfaces will show up with their proper name (i.e. HD Native). The H9000 and other 3rd party HD devices will show up as "HD I/O" with its corresponding device number as shown above.

If a LoopSync connection has been established between your H9000 and another HD device, you will be able to use either the H9000 or the other HD device as the clock source. To select a clock source, click on the *Clock Source* tab and a drop down menu will appear. In the above example, selecting HD I/O #1 or #2 "Internal" will set the HD audio interface as the clock source and the H9000 will be synced to it. Alternatively, selecting HD I/O #3 or #4 "Internal" will set the H9000 as the clock source.

In the example below, the Pro Tools insert **C 5-6** will route to the H9000 HD expansion I/O 5-6.




9K
PROTOOLS C
5
6
1
2
3
4
5
6
7
8
EQ Compressor
Leo to Les
MUTE
BYP
MicroPitch
H3000 MICROPITCH
MUTE
BYP
Tape Echo
TAPE FOR VOCALS
MUTE
BYP
140 EMT Plate
MUTE
BYP

Note that the H9000 displays the expansion card I/O as a single 32-channel device, while Pro Tools divides the I/O into two separate 16-channel devices. In the above example, expansion card I/O 1-16 will appear in Pro Tools as **C 1-16**, while expansion card I/O 17-32 will appear in Pro Tools as **D 1-16**. For example, if you want to route your audio into HD expansion inputs 19-20, they will appear in Pro Tools as **D 3-4**.

7.3.3. Using Multiple Pro Tools | HD Expansion cards

In order to use more than one Pro Tools | HD expansion cards, the clocking must first be setup correctly:

- If there are no other HD devices in the system, LoopSync is not needed.
- Only one card can have Clock Control enabled *AND* it must be the card that Pro Tools selects as the "Loop Master" (Internal/AES/Word Clock).

Let's assume that HD expansion card A is connected to the HD system's Primary port and HD expansion card B is connected to the Secondary port:

- In the Pro Tools *Hardware Setup* page, if HD1 or HD2 is selected as the "Loop Master," then HD expansion card A needs to have Clock Control **ON**.
- If HD3 or HD4 is selected as the "Loop Master," then expansion card B needs to have Clock Control **ON**.
- If another HD device is the "Loop Master," either expansion card can have Clock Control enabled.

A. STARTUP SEQUENCES

The H9000 contains various “Startup Sequences”. By pressing and holding different button combinations while powering up, you can perform the following functions:

A.1. Soft Reset

H9000 - Hold down the **SESSIONS** and **FX CHAINS** buttons while turning on the H9000 by pushing the power button. This will clear the current state of the H9000, but will not enable the Wi-Fi Access Point. This does not affect any of your saved sessions or presets, but will clear the currently loaded session and all settings in Setup View, including your network and authentication settings.

H9000R - Hold down the **NETWORK** button while turning on the H9000R by pushing the power button. This will clear the current state of the H9000, and also enable the Wi-Fi Access Point (as long as a Wi-Fi adapter is plugged into a USB-A port on the H9000). This does not affect any of your saved sessions or presets, but will clear the currently loaded session and all settings in Setup View, including your network and authentication settings.

- The Wi-Fi access point will appear in your computer’s network settings as “H9000” followed by a string of automatically generated characters, or the custom name you have assigned to your unit.
- You will be required to enter an automatically generated password to connect to the Wi-Fi Access Point. This will be your H9000’s serial number in the format “9K-XXXXX”.

⚠ Warning

Turn off the access point after you are finished using it. Otherwise it may interfere with Emote’s connectivity.

A.2. Recovery Mode

Recovery Mode can be used to install an update from a FAT32-formatted USB flash drive. This mode is useful if a problem occurs when updating the H9000’s firmware.

Note

A FAT32-formatted USB drive with a valid firmware installer must be connected to your H9000 before it is powered on to connect to Emote in recovery mode.

To install an update using a FAT32-formatted USB flash drive:

- Download the "USB Installer for H9000", which is available on the Eventide website. If for any reason you need a specific firmware version, please contact support@eventide.com and we will provide you with the correct USB installer.
- Unzip the zip file.
- Inside the folder should be 2 files, "usb-updater.sqz" and "usb-updater-info.xml".
- Copy these 2 files to the top level of a FAT32-formatted USB flash drive.
- Eject the drive from your computer and connect it to a USB port on the H9000.
- *On the H9000:*
  - Hold down the **CHECK** and **X** buttons while turning on the H9000 by pushing the power button.
  - Wait for the H9000 to boot into recovery mode.
  - Follow the on-screen instructions.
- *On the H9000R:*
  - Power on the H9000R while inserting a paperclip into the pinhole that is located to the right of the front panel **NETWORK** button. The **NETWORK** LED will start blinking.
  - If your H9000R is connected directly to your computer or router via ethernet, you will be able to connect to the H9000R RECOVERY device in Emote without any further configuration.
  - If a Wi-Fi adapter is plugged into a USB-A port on the H9000R, it will become visible on your computer's list of available Wi-Fi networks as a RECOVERY Access Point once the **NETWORK** LED stops blinking. Once it becomes visible, connect your computer to it.
  - You will be required to enter an automatically generated password to connect to the recovery device. This will be your H9000's serial number in the format "9K-XXXXX".
  - When the **NETWORK** LED stops blinking, your H9000R should be visible in Emote, displayed as the unit's serial number, followed by the word "recovery". You can connect to the hardware as usual, however the options presented in Emote, once connected, will be different than usual. You can access:

1. The USB updater (if a valid USB device with updater files was connected at boot)
2. Functions for:
    - Clear Current State
    - USB Data Dump
    - Reinitialize (Factory Reset)

- Click the “USB Update” button to install the firmware from the USB flash drive.
- To exit Recovery Mode, simply power cycle the device and it will boot into its normal state. If you’ve performed a firmware update, an option will appear to do this automatically from Emote.

A.3. Factory Reset

> **Warning**
>
> This operation will remove all of your stored Sessions, Algorithms, and Presets.

H9000 - Hold down the **SESSIONS**, **FX CHAINS**, and **ALGORITHMS** buttons while pushing the power button.

H9000R - Push a paperclip into the pinhole while pushing the power button. Connect to the recovery device in Emote and select the “Reinitialize” option.

A.4. USB Dump Data

Insert a FAT32-formatted USB drive into any USB port to backup the contents of your H9000.

H9000 - Hold down the **SESSIONS** button while pushing the power button. Once booted, a message will display on the front panel saying “User data dumped to USB memory.”

H9000R - Push a paperclip into the pinhole while pushing the power button. Connect to the recovery device in Emote and select the “Dump Data” option. A USB installer of the most recent firmware version must be present on the USB stick for this to work in recovery mode.

Note

The USB Dump Data feature can also be accomplished using Emote for either type of H9000. In Devices View (Section 3.3.1), click the *Update* button, and a button to "Dump Data" will appear. This is useful if you'd like to make a backup of your H9000's data before updating its firmware.

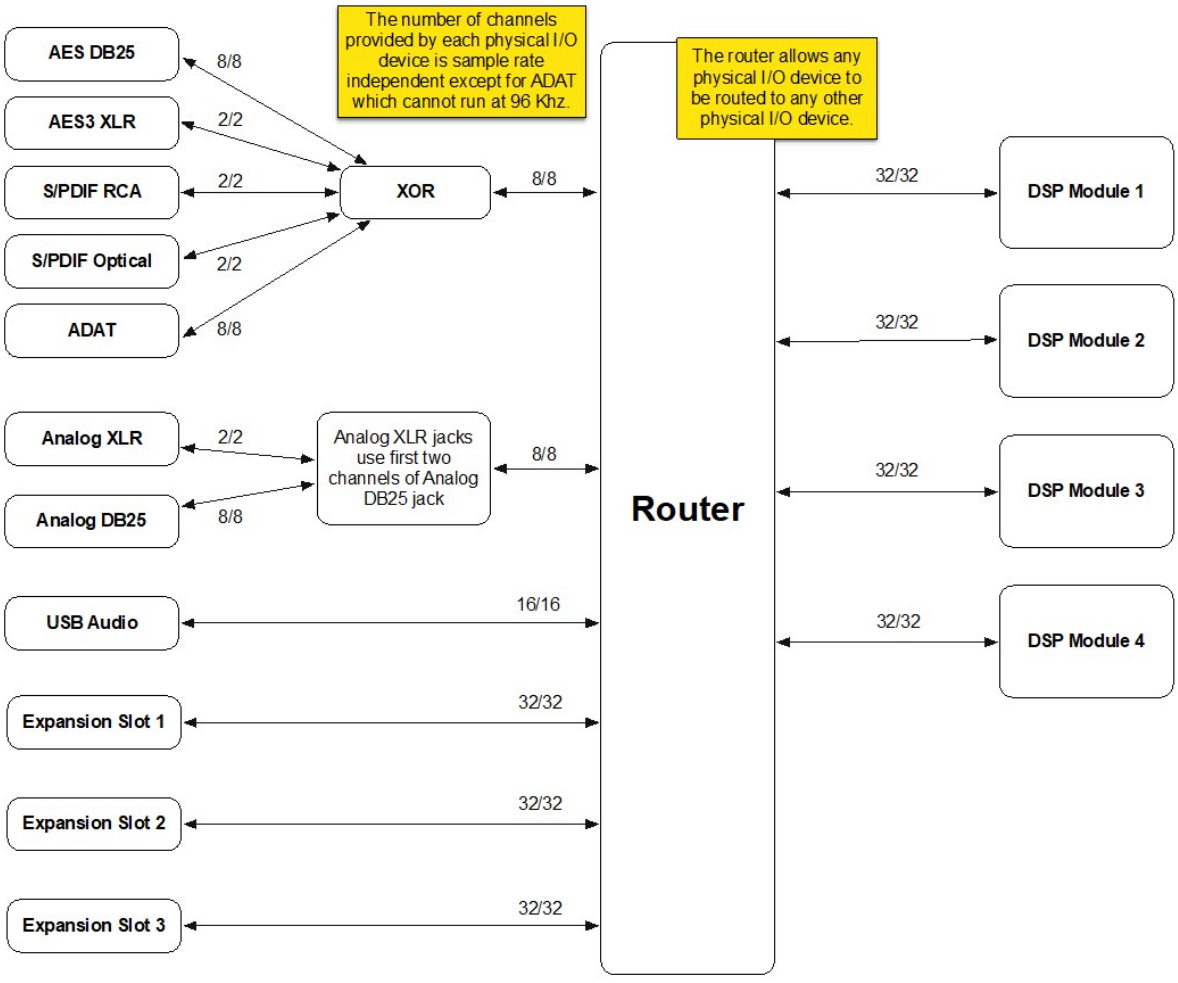
A.5. USB Self-Test Mode

The USB self-test is a way to test various functions and diagnose potential issues on the H9000.

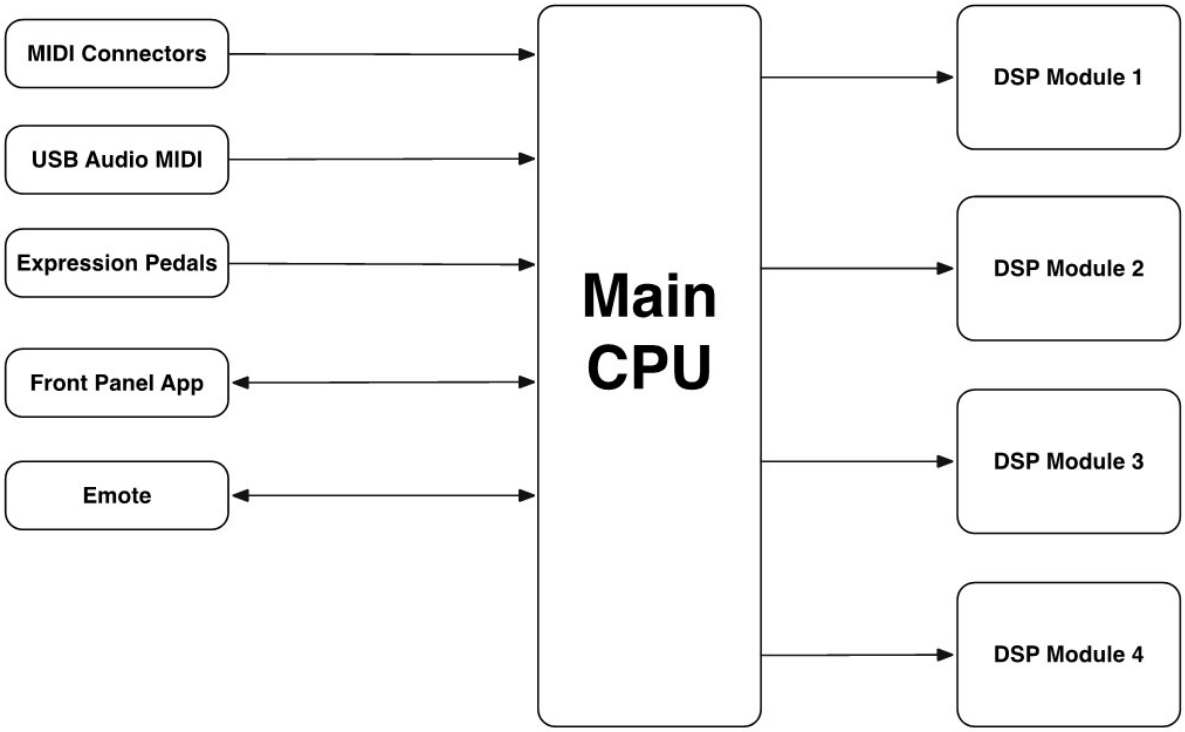
- Download and unzip the USB self-test zip file available on the H9000 Installers webpage: https://www.eventideaudio.com/support/product/9106/installers
- Inside the folder should be 2 files, "usb-self-test.sqz" and "usb-self-test-info.xml".
- Copy these 2 files to the top level of a FAT32-formatted USB flash drive. Make sure there are no H9000 USB updater files on the drive.
- Eject the drive from your computer and connect it to a USB port on the H9000.
- *On the H9000*, power on the unit while holding both the **CHECK** and **X** keys for at least two seconds and wait for it to boot into self-test mode. You will see a progress bar that indicates the tests are being performed.
- *On the H9000R*, power on the unit while pushing a paperclip into the pinhole and wait for it to boot into self-test mode. The **NETWORK** LED will blink until the tests are complete. The LED will be **ON** if all the tests passed, or **OFF** if any failed.
- Once the tests are completed, turn off the unit and remove the USB drive.
- A test log document titled "9K-XXXXX-self-test.log" will now be available for you to view on the USB drive.
- You can email support@eventide.com to submit the test log document.

B. SIGNAL FLOW

B.1. Audio Signal Flow



B.2. Control Signal Flow



C. MIDI FUNCTIONALITY

1. On the H9000, MIDI is always on and received simultaneously by all five USB ports as well as the MIDI-in DIN connector. The USB-A ports can be connected to any MIDI device, and the USB-B port can be connected to any MIDI host.
2. Controller functionalities such as omni-channel, note mode, pitch bend range, etc., are not global. They are made at the controller-mapping level via either the front panel (Section 4.5.1), or Emote (Section 3.3.4.1).
3. MIDI Program changes are handled via Scenes, either on the front panel (Section 4.8), or in Emote (Section 3.3.6).
4. MIDI feedback can be enabled via either the front panel (Section 2.1.8), or Emote (Section 3.3.8).
5. Setting the Tempo Mode to MIDI Clock, either on the front panel (Section 2.1.5), or in Emote (Section 3.3.8), will automatically set the tempo to an incoming MIDI signal (as long as that MIDI signal contains a valid MIDI clock signal). The external MIDI clock signal can arrive either via DIN, or USB-B.

The following action types are available for note-based MIDI controllers:

Velocity	Uses the velocity of the selected note as the control value.
Trigger	Emits a trigger pulse whenever the note is played.
Toggle	Flips between the min and max values when the note is played.
Note Number	Uses the whole range of the keyboard, using the note number as the control value.
Note to Hz	Uses the whole range of the keyboard, and emits a control value equal to the actual frequency of the note being played.

When using Note to Hz, the “Range” dialog contains 2 range knobs:

1. Control Range: Selects the range that controller will respond to from 0-127.
2. Parameter Range: Determines how the control range value is applied to the parameter it’s affecting. For example, you could use half of the Control Range to sweep the full range of a parameter, or you could use the full Control Range to sweep half the Parameter Range.

Warning

Currently, the H9000 does not transmit or respond to any SysEx messages, and does not output MIDI clock information from its USB ports or MIDI-out DIN connector.

Note

Emote's Functions (Section 3.4.1) may also be used to automate FX Chain and Algorithm parameters via a digital audio workstation.

D. SPECIFICATIONS

Audio Specifications	
SNR (A Weighted)	THD
44.1 kHz: >114dB	44.1 kHz: <0.002%
48 kHz: >114dB	48 kHz: <0.002%
96 kHz: >114dB	96 kHz: <0.002%
Analog Audio Connections	
XLR	
Number of Channels	2 inputs, 2 outputs
Pinout	Pin #1 is ground, Pin #2 is +phase (hot), Pin #3 is -phase.
DB-25	
Number of Channels	8 inputs, 8 outputs
Pinout	TASCAM Standard: https://etide.io/DB-25-Pinout
Digital Audio Connections	
USB	
Number of Channels	16 inputs, 16 outputs
Supported Sample Rates	96 kHz, 88.2 kHz, 48 kHz, 44.1 kHz
AES (DB-25)	
Number of Channels	8 inputs, 8 outputs

Digital Audio Connections	
Supported Sample Rates	96 kHz, 88.2 kHz, 48 kHz, 44.1 kHz
Pinout	TASCAM Standard: https://etide.io/DB-25-Pinout
AES (XLR)	
Number of Channels	2 inputs, 2 outputs
Supported Sample Rates	96 kHz, 88.2 kHz, 48 kHz, 44.1 kHz
S/PDIF (RCA)	
Number of Channels	2 inputs, 2 outputs
Supported Sample Rates	96 kHz, 88.2 kHz, 48 kHz, 44.1 kHz
S/PDIF (Optical)	
Number of Channels	2 inputs, 2 outputs
Supported Sample Rates	48 kHz, 44.1 kHz
ADAT (Optical)	
Number of Channels	8 inputs, 8 outputs
Supported Sample Rates	48 kHz, 44.1 kHz
Expansion Cards	
3 expansion slots for additional optional I/O	
Number of Channels	32 inputs, 32 outputs
Supported Sample Rates	96 kHz, 88.2 kHz, 48 kHz, 44.1 kHz
Word Clock (BNC)	
Word Clock Input Termination	75 Ohms, switchable

Digital Audio Connections
Synchronization Sources
Internal, Word Clock, AES XLR, AES DB-25, S/PDIF, ADAT, Expansion Card

Control Connections
Remote Control Inputs
Up to four foot pedals, or four 5v control voltage sources, or six foot switches, or combination
2 ¼" TRS inputs. The sleeve is ground reference, the ring is +5 volts (source), and the tip is an analog signal from 0 to 5 volts. Connect either foot switches, foot pedals, or control voltage sources.
Remote Control Outputs
2 isolated relay closures, rated at 30Vdc 1A. Relay #1 is connected between ring and sleeve, while Relay #2 is connected between ring and tip. These connections are electrically isolated from the H9000.
MIDI Connections
5-pin DIN Input, Output, and THRU
USB A and USB B

Physical Dimensions	
Size (2U rack space)	Net Weight
3.5h x 19w x 12.5d in	14 lbs
8.9h x 48.3w x 31.7d cm	6.4 kg

All specifications are subject to change or improvement at any time without notice or obligation. Please visit https://etide.io/H9000Specs *for updated product specifications.*

This page was intentionally left 99.88% blank.

E. TECHNICAL SUPPORT

H9000 Youtube Tutorials

This playlist includes helpful tutorials for setting up and using your H9000: https://etide.io/H9000Tutorials

Eventide Rackmount Forum

The official Eventide forum for the H9000 and other rackmount processors. New H9000 and Emote releases will be announced here periodically: https://etide.io/rackmount-forum

VSig Forum

Support for VSig and the development of H9000 algorithms using VSig: https://etide.io/vsig-preset-forum

H9000 Official Facebook Group

A group where H9000 users can interface with each other and members of the Eventide team: https://etide.io/FB-H9000

H9000 FAQ

Get answers to frequently asked questions and get the most out of your H9000: https://etide.io/H9000FAQs

Contact Eventide Support

If you need technical assistance or product repair, you can email support@eventide.com to submit a ticket to our support department. Please include a detailed report of your problem including your H9000's serial number, firmware version, and what version of Emote you are using. Most inquiries are answered within 24-48 business hours. Note, we are closed on weekends.

This page was intentionally left 99.88% blank.

F. ALGORITHM DEVELOPMENT

Algorithms can be developed for the H9000 using Eventide's VSig3, and Cycling 74's RNBO and Gen.

6.1. VSig

VSig is a visual editor used to create custom algorithms for the H9000. VSig utilizes a collection of precompiled code "blocks" known as modules, which are connected directly to one another to create patches. Algorithms can be uploaded directly and immediately to the H9000, with user definable UI objects (such as knobs and faders) which are viewable on the H9000's front panel, or Emote.

6.1.1. Installation/Setup

VSig3 is available for download on the H9000/H9000R downloads page on the Eventide website.

To be able to download VSig, you must register your H9000 or H9000R serial number with your Eventide account. One registered, the VSig installer will become available for download.

To connect VSig to your H9000/H9000R, open VSig settings by navigating to File > Settings. An H9000 connection can be established using either the device's IP Address or name. By default, the device name is its serial number in the format 9K-XXXXX. Both the device's name and IP address are viewable and configurable in Emote.

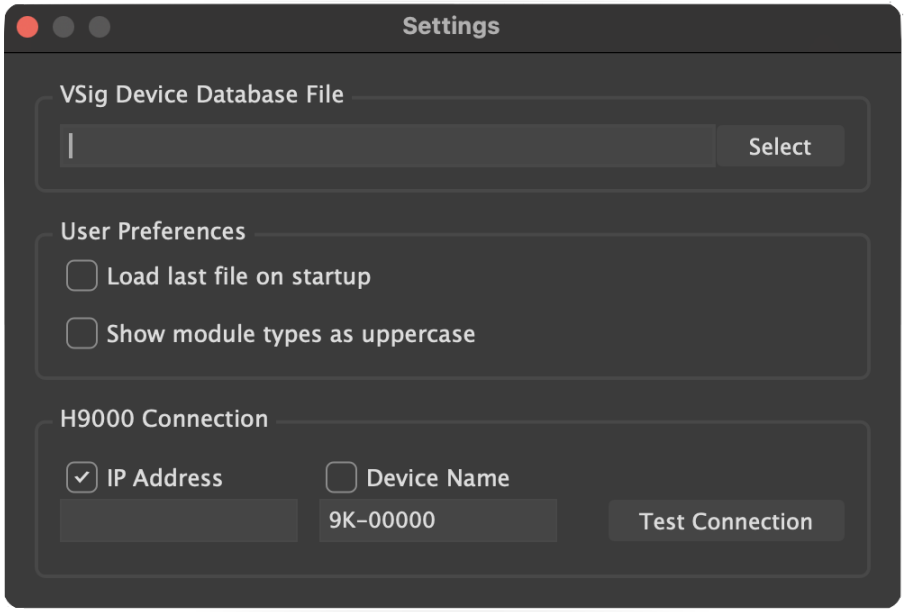


Fig. 6.1 **VSig Settings**

Once connected, you can begin uploading/downloading algorithms to/from the H9000.

6.1.2. Connection Types

There are 3 basic connection types used to create algorithms in VSig:

- Signal (green) - used for sending audio signals
- Control (blue) - used for sending control signals; often involves using interface modules to control parameters of other modules
- User (pink) - used for GUI purposes; allows for the creation of menu pages, which display interface objects (such as knobs, faders, monitors, etc.) to users via Emote

6.1.3. Tool Bar Functions

Fig. 6.2 **VSig Toolbar**

1. Undo
2. Redo
3. Zoom in
4. Reset to default zoom
5. Zoom out
6. Zoom to fit
7. Hide/show signal connections
8. Hide/show control connections
9. Hide/show user connections
10. Increase/decrease opacity of modules
11. Download an existing algorithms VSig file from H9000
12. Upload algorithm either to first slot on FX chain, next empty algorithm location, or an algorithm location of your choice
13. AutoLayout algorithm

6.1.4. Modules

Modules are precompiled “blocks” of code which act as the building materials for all VSig algorithms. Modules are able to send and receive signals to and from one another by clicking and dragging from one module to another to establish connections.

Each module has documentation containing details about its functionality, inputs, outputs, and parameters. To access this, right click on a module, and select “Show Module Documentation.”

6.1.4.1. Module Types

- Bridge – convert between signal and control connections
- Controlmath – performs mathematical operations on control signals, including basic arithmetic, comparison, conversion, logical operators, etc.
- Controlprocess – provides further methods for manipulating/operating on control signals

- Delay – various forms of delaying/phase shifting an audio signal; includes modulatable delays, all pass filters, comb filters, etc.
- Detector – read various quantities associated with audio signals and output them as a signal; can detect pitch, peak amplitude, envelope, etc.
- Dynamic – used to modify gain/amplitude of an audio signal; includes compressors/expanders, gates, duckers
- External – used for accepting signals from external control surfaces, such as MIDI interfaces or expression pedals
- Filter – used for altering the frequency or phase response of an audio signal; includes standard filter/EQ modules which attenuate harmonics, in addition to objects which add harmonics (distortion)
- Interface – used to create GUI objects that the user can interact with via the H9000, or within Emote if using H9000R
- Math – performs mathematical operations on audio signals; includes many of the same modules as Controlmath, with some additions such as gain
- Miscellaneous – self explanatory
- Mixer – used for various signal routing purposes; includes standard mixers, in addition to switches, crossfaders, panners, etc.
- Oscillator – generate waveforms, including envelopes, LFOs, impulses, noise, or other audible waveforms
- Pitchshift – used for detuning/pitch shifting audio signals; includes several options depending on the required amount of pitch shifting, whether or not audio needs to be shifted diatonically, etc.
- Reverb – includes “plug and play” reverb modules (Reverb_2016, stereo_room, and reverb_a,b,c,d) which can be used on their own without additional modules, in addition to modules which can be used to build reverb algorithms from the ground up.

6.1.4.2. Head Module

The head modules appear on the left and right of the editor UI, and are automatically generated when you create a new patch. It is used to route audio and control signal in and out of a VSig patch. It looks like this:

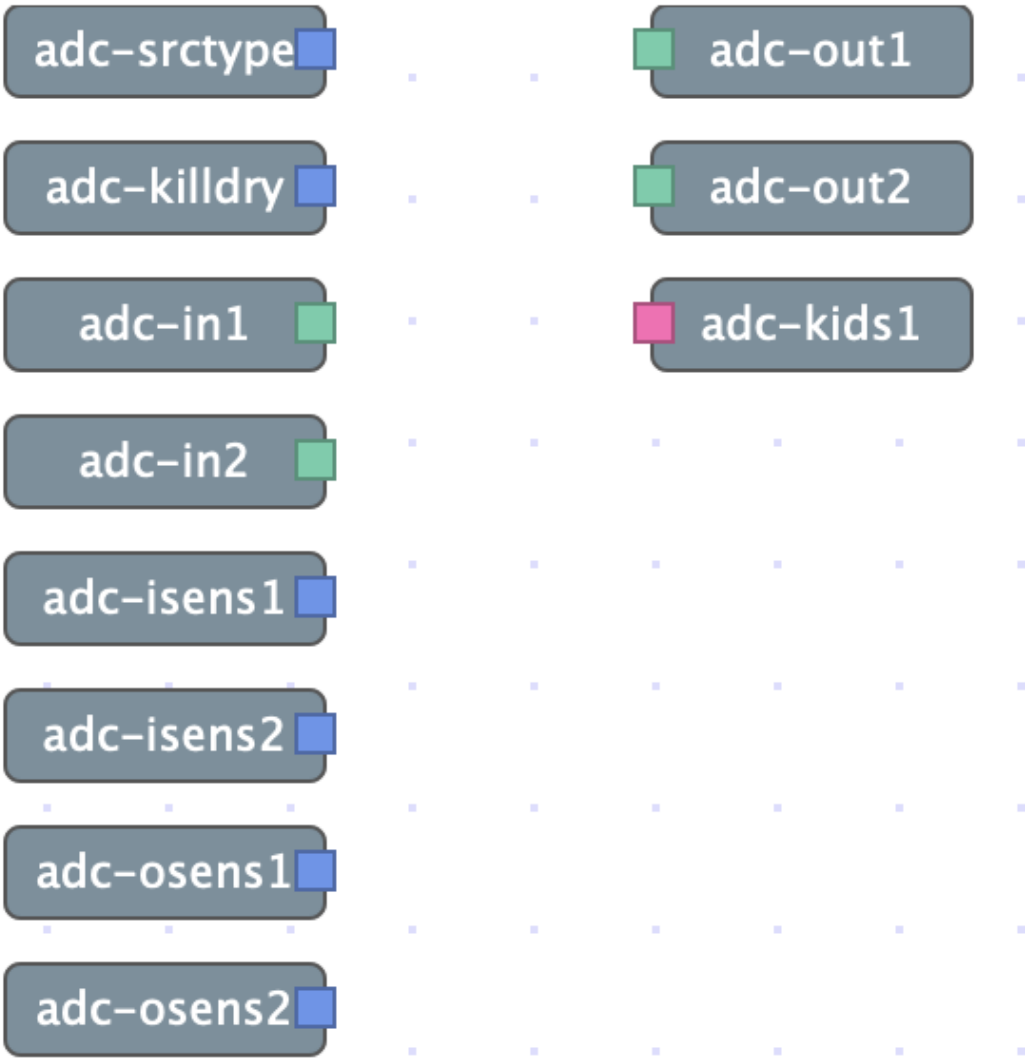


Fig. 6.3 **Head modules generated automatically when creating a new VSig file**

The list of head submodules is as follows:

- adc-scrtype: Gives the current input source type [1=guitar, 2=bass, 3=lead, 4=sub]
- adc-killdry: Gives the current killdry status
- adc-in1, adc-in2: Audio input from corresponding adc channels (more adc-ins can be added in the module editor)
- adc-isens1, adc-isens2: Gives the connectivity status of the corresponding adc input
- adc-osens1, adc-osens2: Gives the connectivity status of the corresponding adc output

- adc-out1, adc-out2: Audio output to corresponding adc channels (more can be added in the module editor)
- adc-kids1: Output for UI object groups to emote (more can be added in the module editor.

6.1.4.3. Module Editor

The module editor is displayed on the right-hand side of the screen whenever a module is selected, and displays important information regarding parameters, connections, and objects associated with a module.

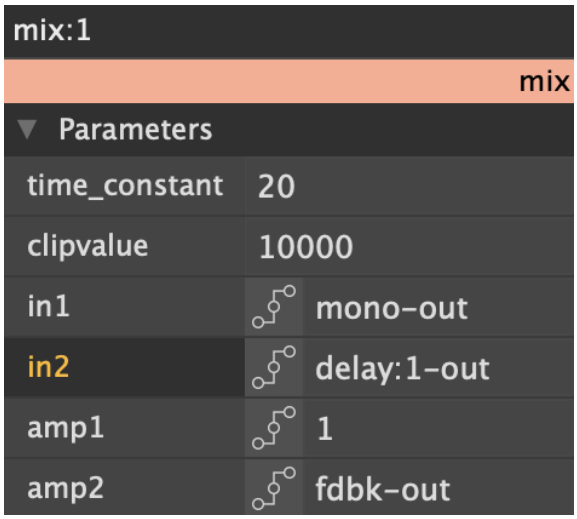


Fig. 6.4 **Editor page for a mix module. All parameters are shown, including signal inputs and their associated amplitudes.**

The module editor can be used to manually define parameters via text entry, or may show how a parameter is being controlled by an interface object. In the example above, amp1 is manually defined to have a value of 1, whereas amp2 is being controlled by the output of an interface module with the name "fdbk."

in1 and in2 are both signal inputs which are receiving audio signals from other modules. in1 is receiving signal from a module named mono, while in2 is receiving signal from a module named delay:1.

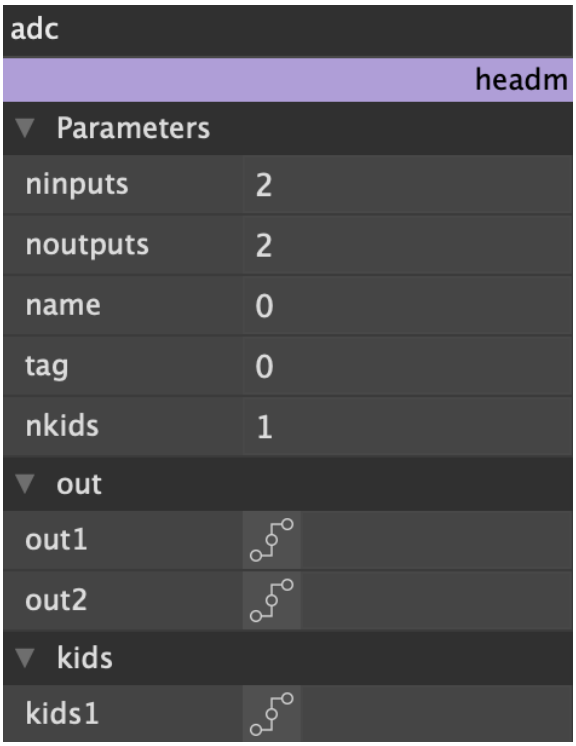


Fig. 6.5 **The module editor also displays important information regarding the head module and its associated submodules. Clicking on any head submodule opens the editor pictured above.**

6.1.4.4. Interface Modules and Creating UI

Interface modules are somewhat unique in that they only utilize control and user connections, never audio signal connections. Control connections are used to send control signals which target parameters of other modules. User connections are used to make interface modules visible within Emote, allowing for the creation of custom GUI.

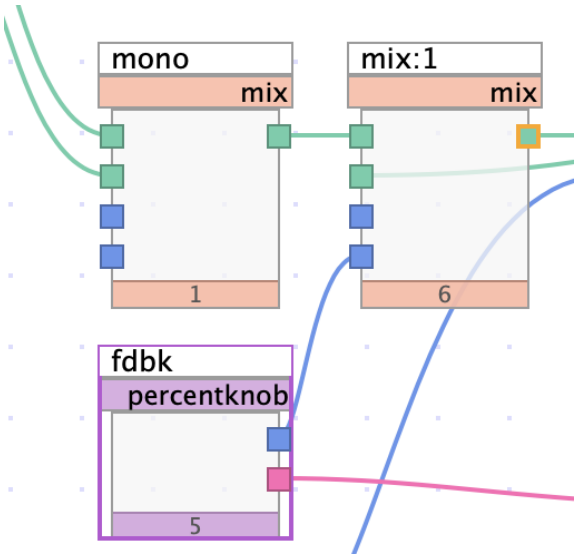


Fig. 6.6 ***Percentknob* module connected to *mix* module via control connection**

A small section of a patch shown above demonstrates the use of control connections. The output of the knob module labeled “fdbk” is connected to the amp2 input. When viewable in Emote, the knob’s position at any given time corresponds to a value which determines the amplitude level of input2. As mentioned above, this connection can also be seen in the module editor.

The range of values accessible by an interface and its resolution are both user definable. Min and max parameters are self-explanatory, while resolution determines the smallest possible discrete value by which an interface object can change.

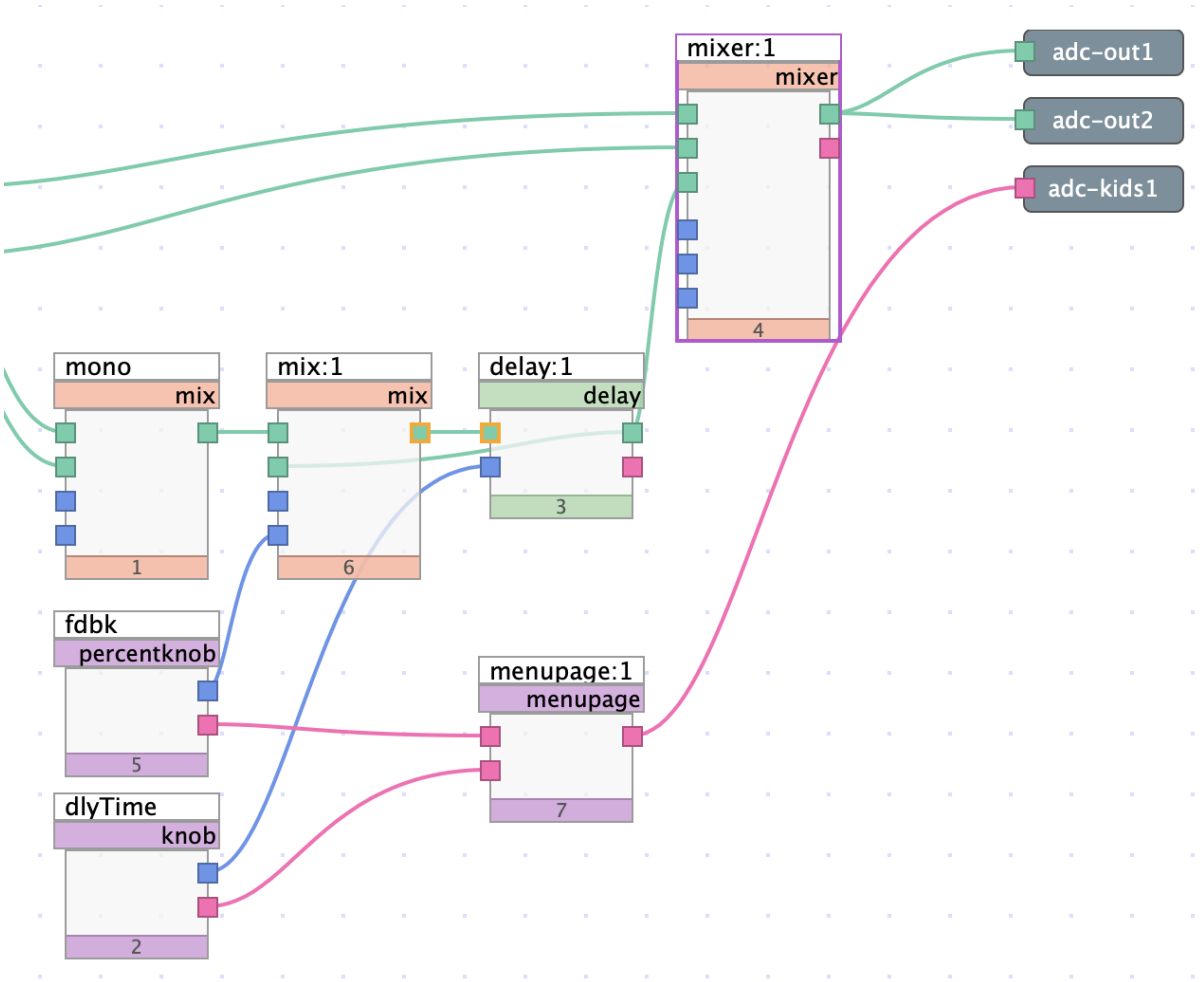


Fig. 6.7 ***Percentknob* and *knob* module both connected to *menupage* module via user connections**

For ANY interface module to be visible in Emote, it must be connected to a menupage module, which must subsequently be connected to adc-kids. The sample patch above shows two knob modules (fdbk and dlyTime) being connected as described above.

The module editor for most interface objects includes a "statement" field. Text entered in this field will be displayed to the user alongside the UI object in Emote. *To display values which change in real time as you interact with the UI, the following syntax must be used:*

- Changing values are wrapped by the % and f characters. Text to be displayed *under* a knob should be placed before a colon (:).
- Between the % and f should be a decimal number. The whole number portion of your decimal indicates how many digits will be displayed before the decimal point. The decimal portion indicates how many digits will be displayed after the decimal point.

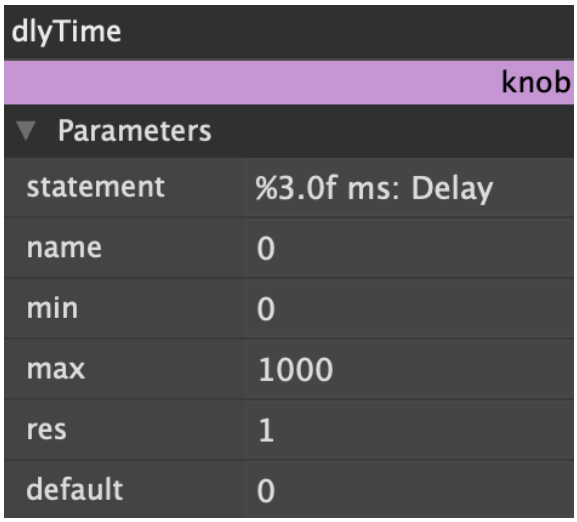


Fig. 6.8 ***Knob* module utilizing %f syntax in its *statement* field**

- The example above will show a knob labeled "Delay," and show the delay value, in ms, at each knob position, with up to 3 whole number digits and 0 decimal digits. *(The %f syntax exclusively affects how values are displayed. It is possible to have an interface module which has higher resolution than is shown to the user)*
- Text outside of the %f indicators will be displayed as normal, unchanging text

UI design is explored in greater detail in Vsig Beginner Tutorial #4 - Intermediate Menu Design.

6.1.5. Downloading and Uploading Algorithms

To download an algorithm from the H9000 into VSig, click the icon with the downward arrow inside the cloud. Once downloaded, the user will be able to see exactly which modules are used on a preexisting algorithm and how they are connected. **When learning VSig, downloading and reverse-engineering preexisting algorithms is a very effective way to begin learning how certain modules work and how modules may be connected.**

To upload an algorithm from VSig to the H9000, click the icon with the upward arrow inside the cloud. Algorithms can be sent directly into the FX chain, or can be stored within Emote in any empty algorithm slot. Once uploaded, algorithms can be tested immediately.

6.1.6. Basic Delay Algorithm

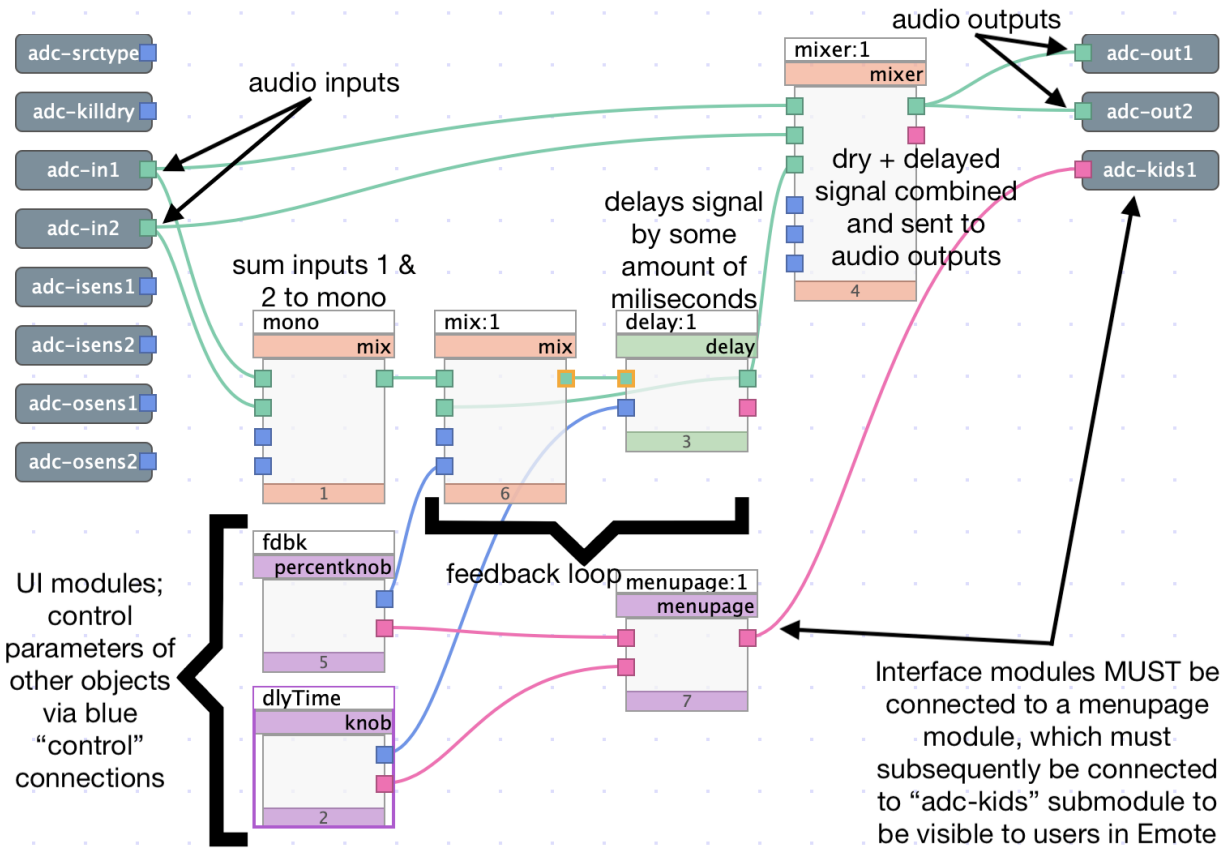


Fig. 6.9 **Simple delay algorithm**

The algorithm shown above is a simple delay, intended to demonstrate how modules may be connected to one another to create audio effects. The audio inputs, labeled *adc-in1/2*, are first split into a dry path and a delay path. The dry path is routed straight to a *mixer*, which is sent directly to the audio outputs.

The delay path first utilizes a *mix* module which sums the two inputs into a singular output. The signal is then sent to another mix module followed by a *delay* module. The combination of the *mix* and *delay* modules not only allows the signal to be delayed by a user definable time, in milliseconds, but also feeds the delay signal back to its input to create a feedback loop. The *delay* output is also routed to the mixer, combining it with the dry signal.

6.1.7. Legacy Support

The current version of VSig, VSig3, is only available for H9000 and H9000R owners and is only compatible with those devices. Several legacy rackmount devices, however, continue to offer support for VSig2. These devices include:

- H8000 family (H8000, H8000A, H8000FW)
- H7600
- Orville
- DSP family (DSP7000/7500, DSP4000B+)

Those who would like to develop algorithms for any of the above devices may refer to the following resources:

- VSig2 product page
- Programming Manual for H8000, H7600, Orville and the DSP7000 families of Harmonizer® Brand Effects Processors

VSig2 functions quite similarly to VSig3 with some variance. The minutiae of this variance will be revealed to those who choose to dive deep into VSig2 development, but several key differences are as follows:

- VSig2 is only available on Windows machines (Mac users may find success with virtual machines, but Eventide is unable to assist in replicating such a setup)
- VSig2 requires that a device connection be established via a MIDI interface or serial connection, as opposed to VSig3's WIFI/Ethernet connection
- VSig2 utilizes a tool called the "VSig Specifier Display," which is an equivalent to VSig3's module editor. The VSig Specifier Display can be opened by double-clicking on a module.
- To add modules to your VSig2 patch, you must use the "Add Module" command in the "Edit" menu. Modules are slightly more "hidden" than in VSig3, which constantly displays the module library on the left side of the screen.
- Legacy rackmount gear does not have companion software like H9000's Emote. Because of this, UI should be designed for the device's built-in display (see pages 45-47 of the Programming Manual for in depth assistance with menu design)
- VSig2 includes a "supermodule" function, allowing several modules to be subsumed into a single supermodule which can be saved to the library (pages 57-61 of the Programming Manual)

Any other subtle variations between VSig2 and VSig3 will likely be revealed through use, or by reading the Programming Manual. Despite the differences listed above, core functionality is nearly identical between VSig2 and VSig3, making it easy to begin developing with one if you are already familiar with the other.

6.2. VSig Tutorials

6.2.1. VSig Beginner Tutorial #1 - Simple Gain

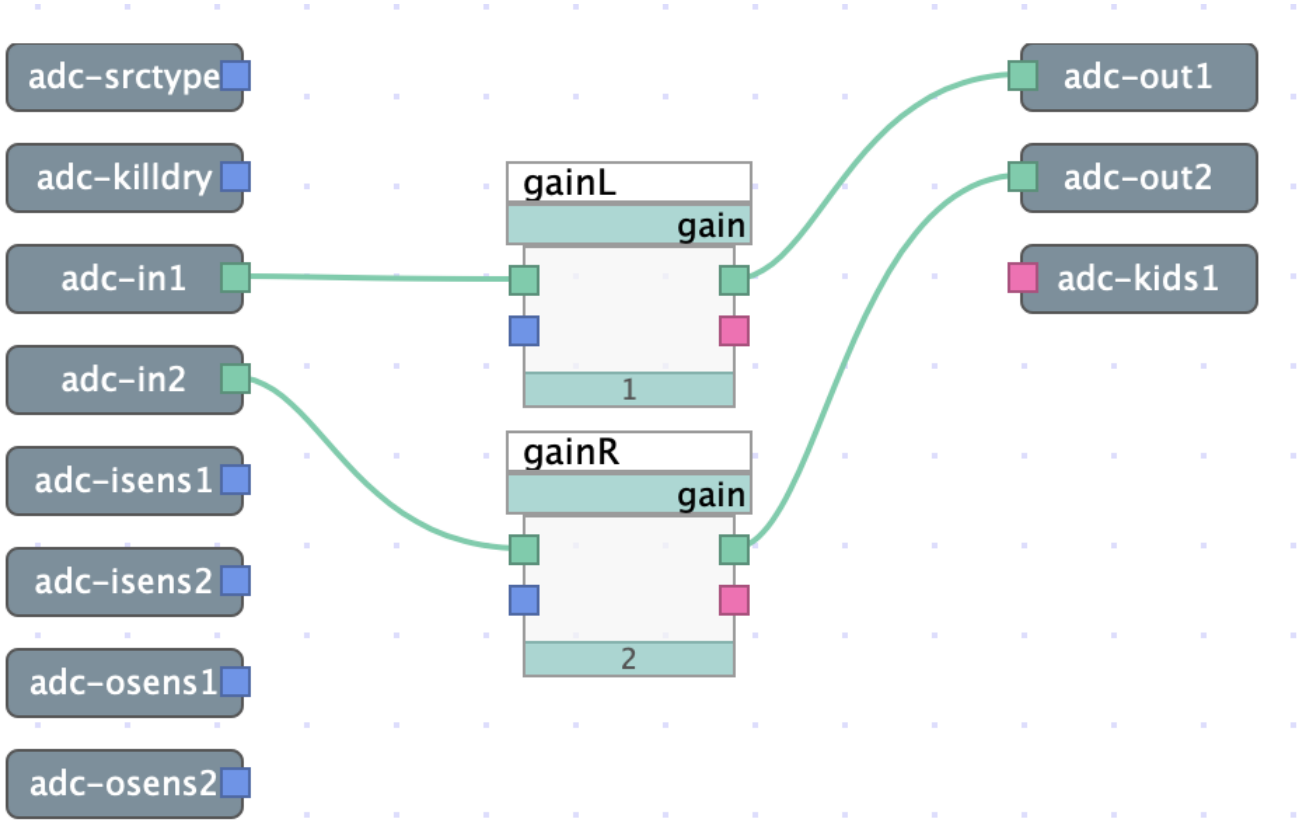


Fig. 6.10 **Simple gain algorithm with no user control**

The above patch is designed to show how audio signal is routed in VSIG using two *gain* modules. By clicking and dragging from a green signal output to an input, a connection is created which allows audio signal to flow. In this case, the *adc-in* outputs are each connected to the input of a *gain* module, and the

output of each *gain* module is subsequently connected to the input of each *adc-out*. Thus, a full signal path is created.

Download the sigfile here: Simple Gain.

6.2.1.1. Gain Module pt.1

Each *gain* module contains a signal input and output, a control input which allows the amount of gain to be controlled by interface objects, and a user output. For the sake of simplicity, in this case the amount of gain is set manually via text entry and is fixed.

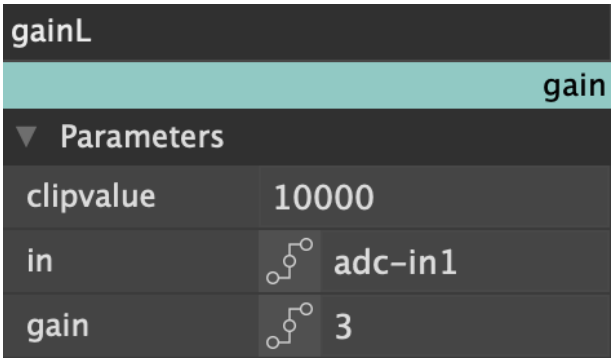


Fig. 6.11 **Left channel gain**

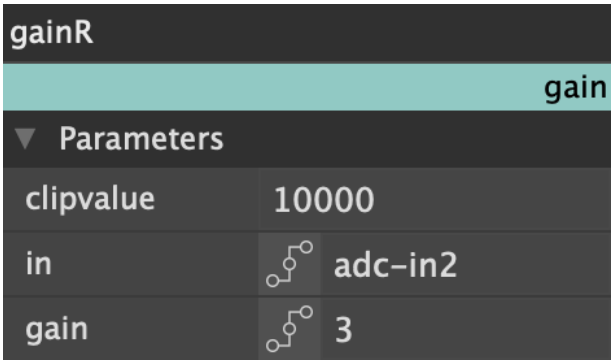


Fig. 6.12 **Right channel Gain**

Both *gain* modules have their gain manually set to 3 dB, providing a fixed amount of gain for each channel which cannot be changed by the user.

Upon uploading the algorithm to Emote, no controls will be displayed to the user, but 3 dB of gain will be applied to both channels. This algorithm is not intended to be complex, but to demonstrate in simple terms how audio signals travel within the VSIG patching environment.

6.2.2. VSig Beginner Tutorial #2 - Simple Gain with UI

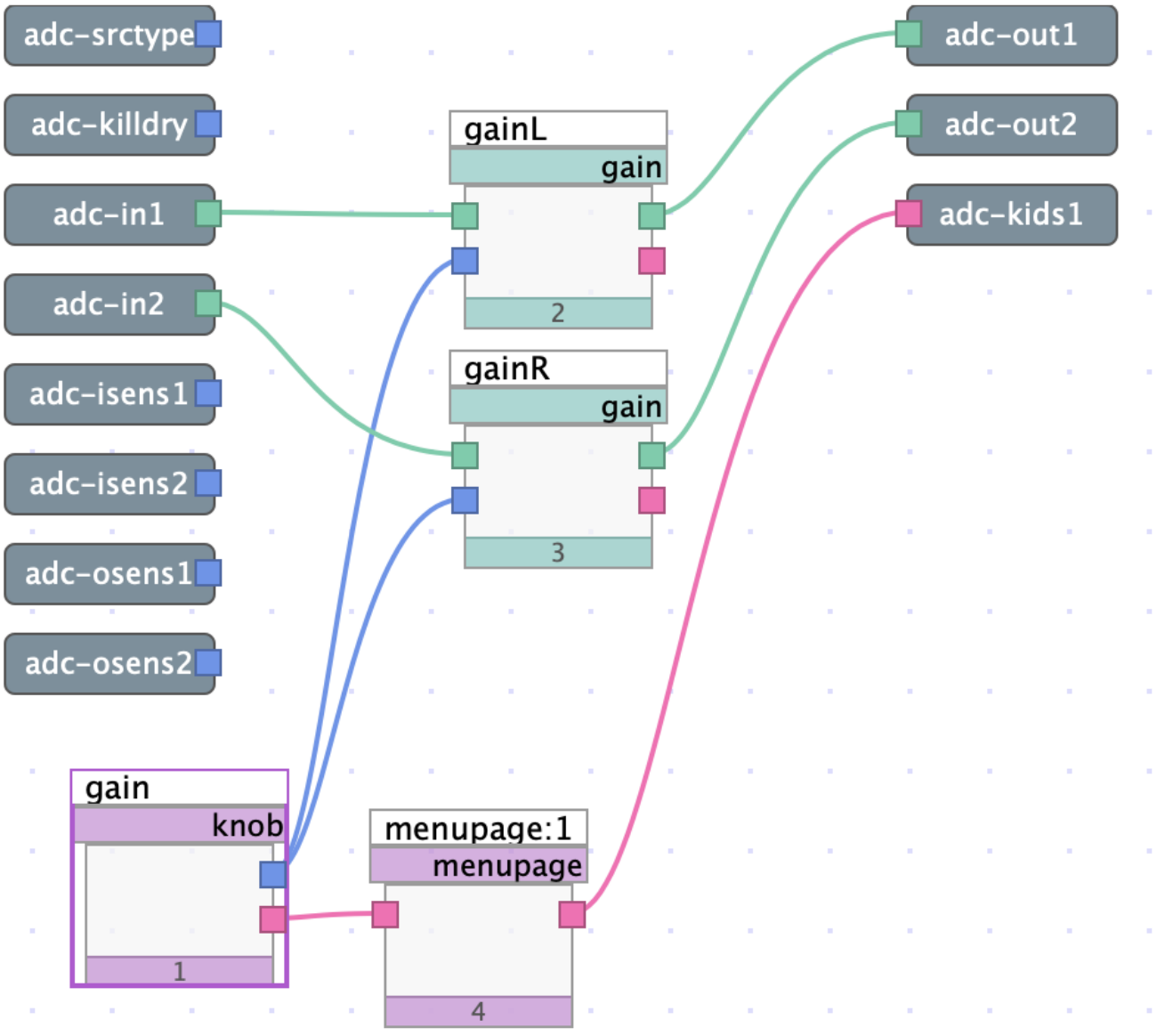


Fig. 6.13 **Simple gain algorithm, with the addition of a *knob* and *menupage* to allow for user control**

The above patch is an expansion upon Beginner Tutorial #1, adding a *knob* to control the gain, and a *menupage* which displays the knob to the user. The addition of interface objects demonstrates how to properly use control and user connections.

Download the sigfile here: Simple Gain with UI.

6.2.2.1. Gain Module pt.2

Signal connections are unchanged from Beginner Tutorial #1, but control inputs are now being utilized. Instead of a predetermined gain value, a *knob* now sends a variable value to the *gain* module via the control input.

6.2.2.2. Knob Module

The knob module contains one control output, which is used to control the gain modules.

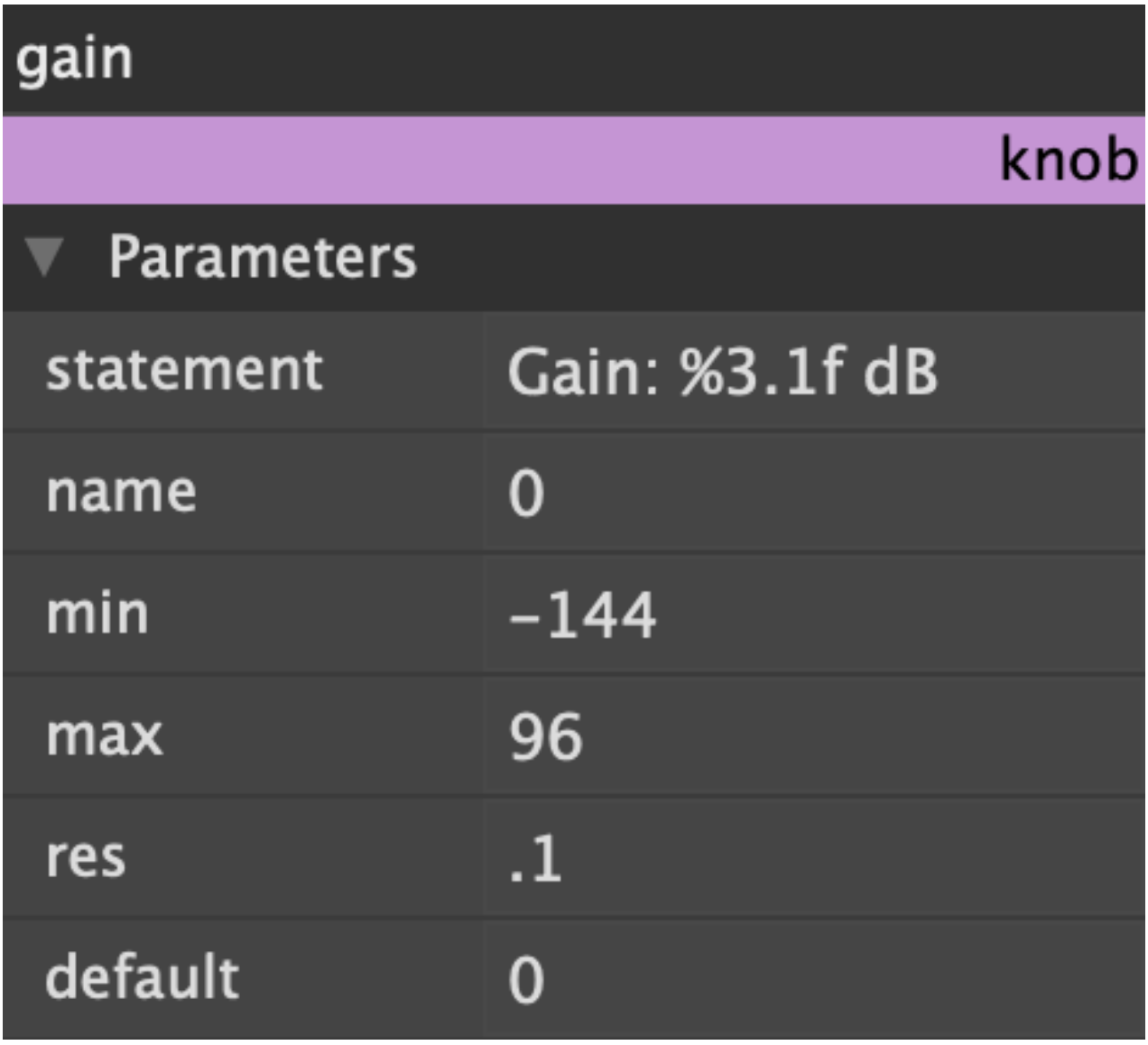


Fig. 6.14 **Module editor for *knob* module, controlling gain**

The above photo shows the minimum and maximum values accessible by the *knob*. These values correspond directly to the min and max values specified by the *gain* module itself, although this is not a requirement.

Field	gain
Description	gain
Default	0
Minimum	−144
Maximum	96

Fig. 6.15 **Minimum and maximum values accessible by *gain* module, viewable in the module editor**

Additionally, the res value determines the increment by which the gain will increase or decrease when interacting with the knob.

The "%...f" syntax is used to display changing numerical values to the user, and is described in more detail in Vsig Beginner Tutorial #4 - Intermediate Menu Design.

6.2.2.3. Menupage Module

The *menupage* module contains a user input and output. The *knob* is connected to the user input, and the user output is connected to *adc-kids*, both of which are required to display the knob as a UI object in Emote.

Upon exporting the algorithm to Emote, a single knob will be displayed to the user labeled "Gain," allowing users to manually control gain between −144 and 96 dB. The numerical value which corresponds to the knob's position is also displayed to the user, showing exactly how much gain is being provided at a given moment.

6.2.3. VSig Beginner Tutorial #3 - Filters

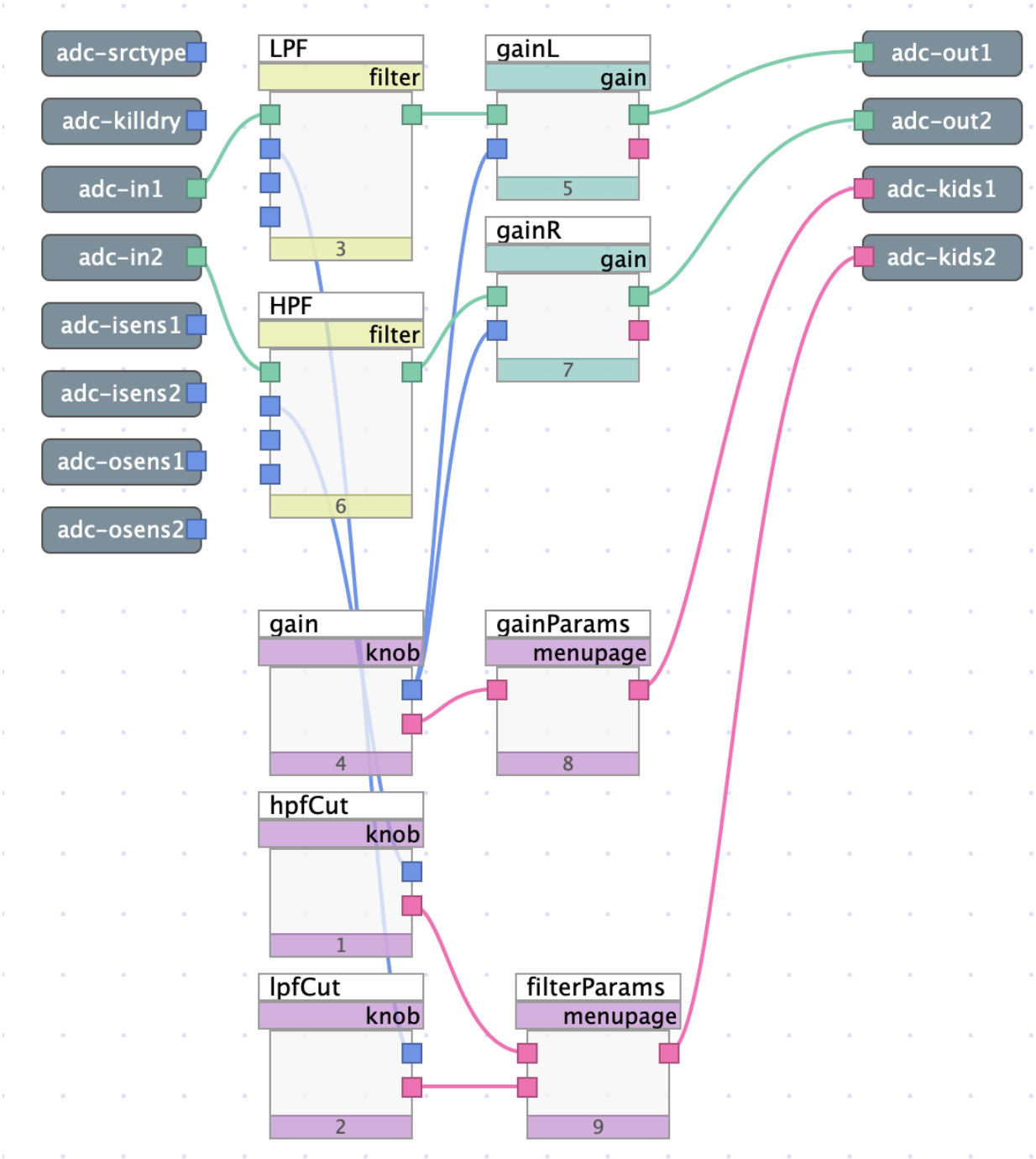


Fig. 6.16 **Gain algorithm with filters and interface objects to control cutoff frequency**

The above algorithm continues to expand upon the first two tutorials, adding two *filter* modules and some additional interface objects. Using *filters*, this patch will demonstrate not only how each input can be processed differently, but also how to create several *menupages* for a single algorithm.

Download the sigfile here: Simple Gain with Filters.

6.2.3.1. Filters

The signal outputs of *adc-1* and *adc-2* are routed to the input of two *filter* modules, titled LPF (Low Pass Filter) and HPF (High Pass Filter) respectively.

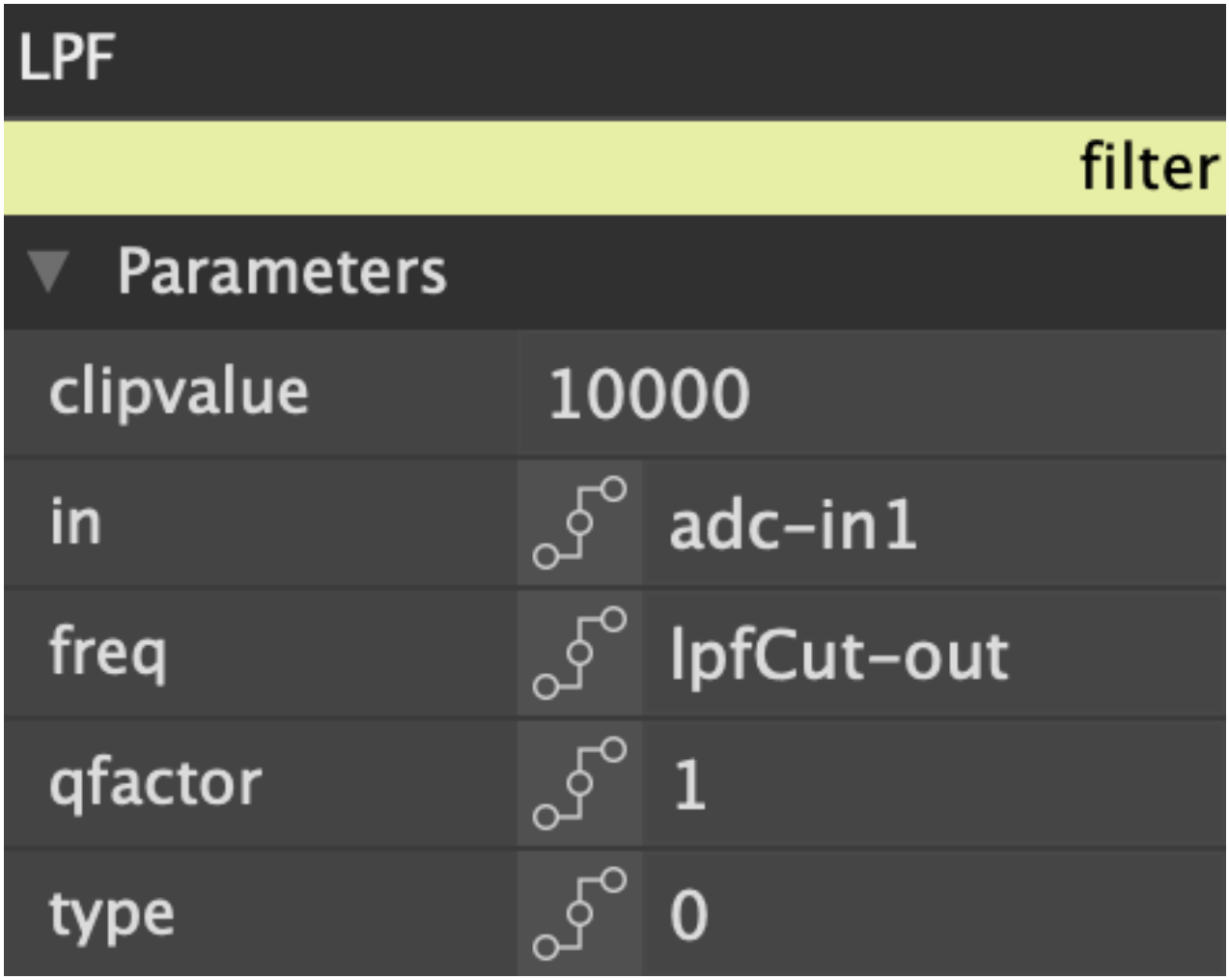


Fig. 6.17 **Module editor for Low Pass Filter**

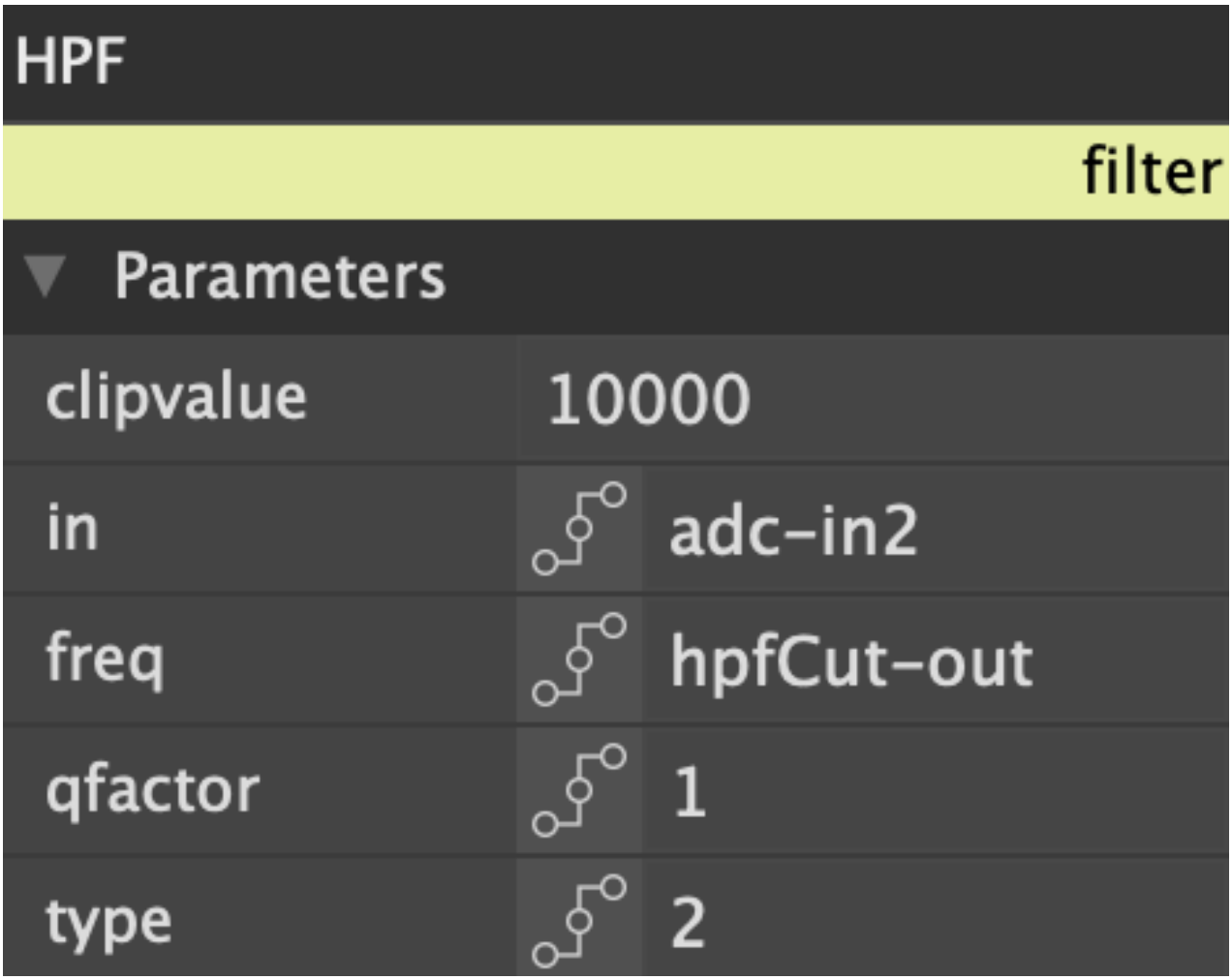


Fig. 6.18 **Module editor for High Pass Filter**

Looking into the module editor, each *filter* has a different value for the "type" parameter, which specifies filter type (low pass, band pass, high pass, or notch). "0" specifies a low pass filter, "1" specifies a band pass filter, "2" specifies a high pass filter, and "3" specifies a notch filter. Channel 1 is routed through a low pass filter, whereas channel 2 is routed through a high pass filter.

6.2.3.2. Interface

Each *filter* has a *knob* connected to its frequency control input, allowing for individual control over the cutoff frequency of each filter. Although these *knobs* could've been connected to the original menupage module by adding more user inputs, in this case a new menupage was added to create a more organized UI.

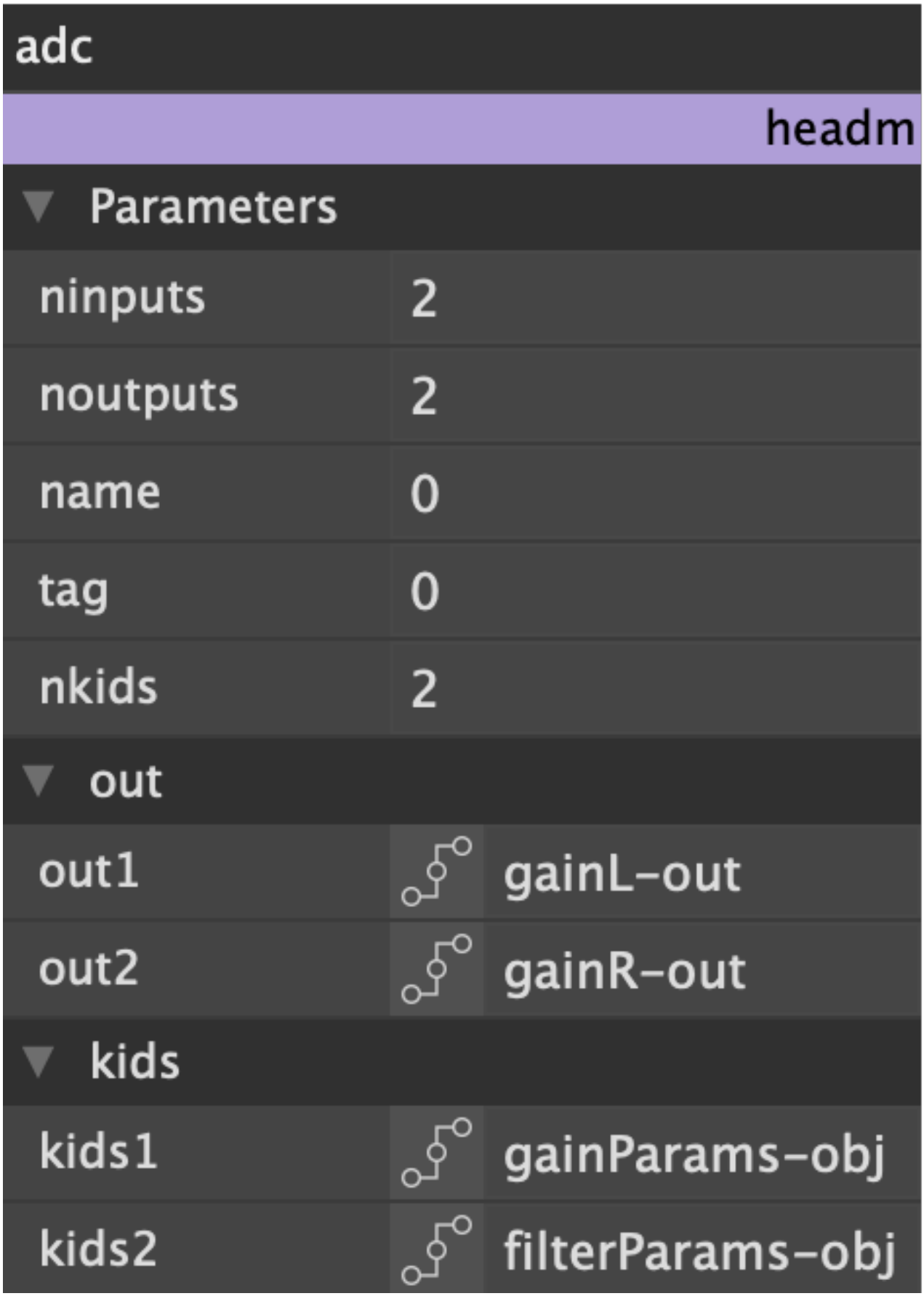

adc
headm
Parameters
ninputs 2
noutputs 2
name 0
tag 0
nkids 2
out
out1 gainL-out
out2 gainR-out
kids
kids1 gainParams-obj
kids2 filterParams-obj

Fig. 6.19 **Head module editor, with *nkids* set to 2**

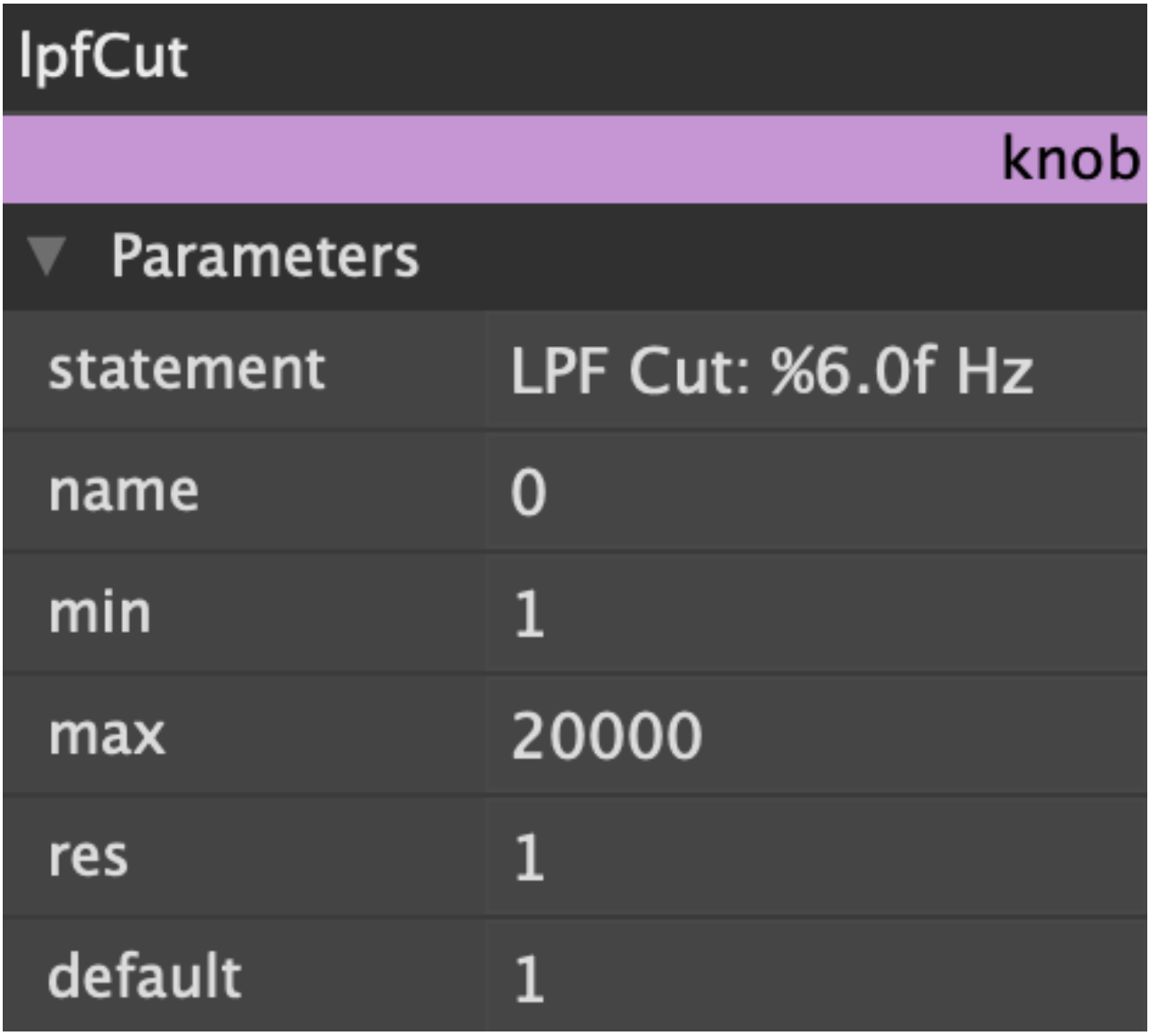


Fig. 6.20 ***Knob*** **module controlling Low Pass Filter cutoff frequency**

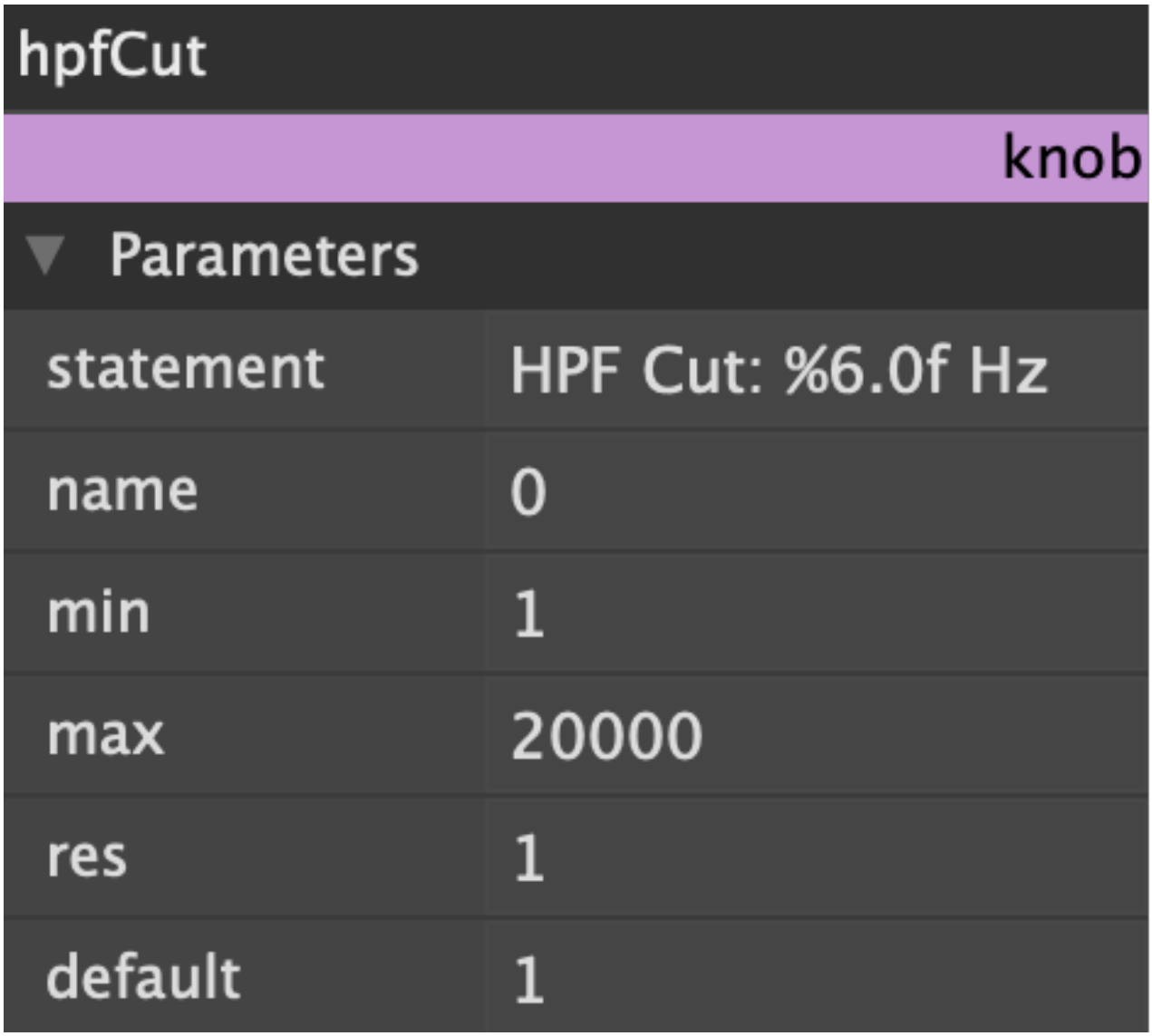


Fig. 6.21 ***Knob*** **module controlling High Pass Filter cutoff frequency**

Each *menupage* requires its own *adc-kids* connection. To add more *adc-kids*, click on any head submodule to open the head module editor, and specify the number of *adc-kids* in the nkids field (in this case 2).

Upon uploading the algorithm to Emote, two separate menus will be displayed, labeled “Gain” and “Filter.” The Gain menu will have one knob labeled “Gain,” and the Filter menu will have two knobs labeled “LPF Cut” and “HPF Cut.” The low pass filter will only affect the channel 1 input, and the high pass filter will only affect the channel 2 input. The gain knob will affect both channels.

6.2.4. VSig Beginner Tutorial #4 - Intermediate Menu Design

This tutorial will focus on making more intricate user menus using VSig by building a chorus effect. The basic effect idea is as follows:

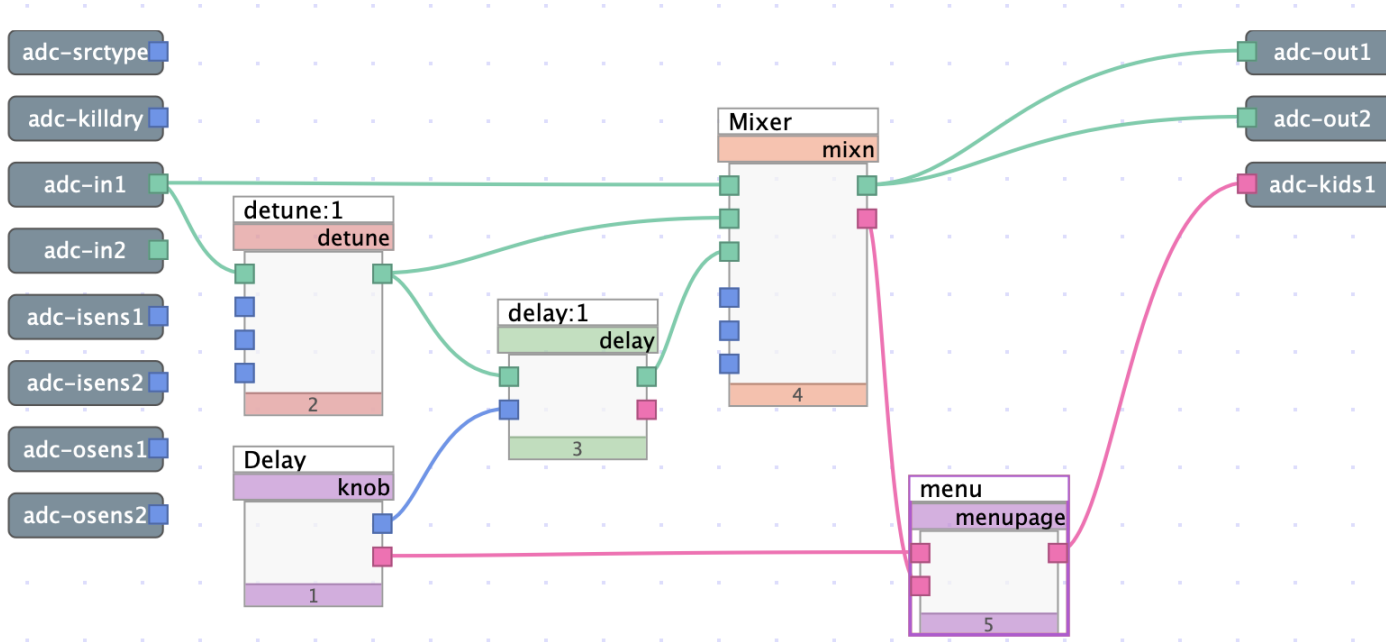


Fig. 6.22 **Simple chorus algorithm with one *knob***

This is a simple chorus effect with a one-knob design. In this case, the *knob* controls the delay time used in the chorus effect. When looking at the available parameters for the *knob* module, we can see the following:

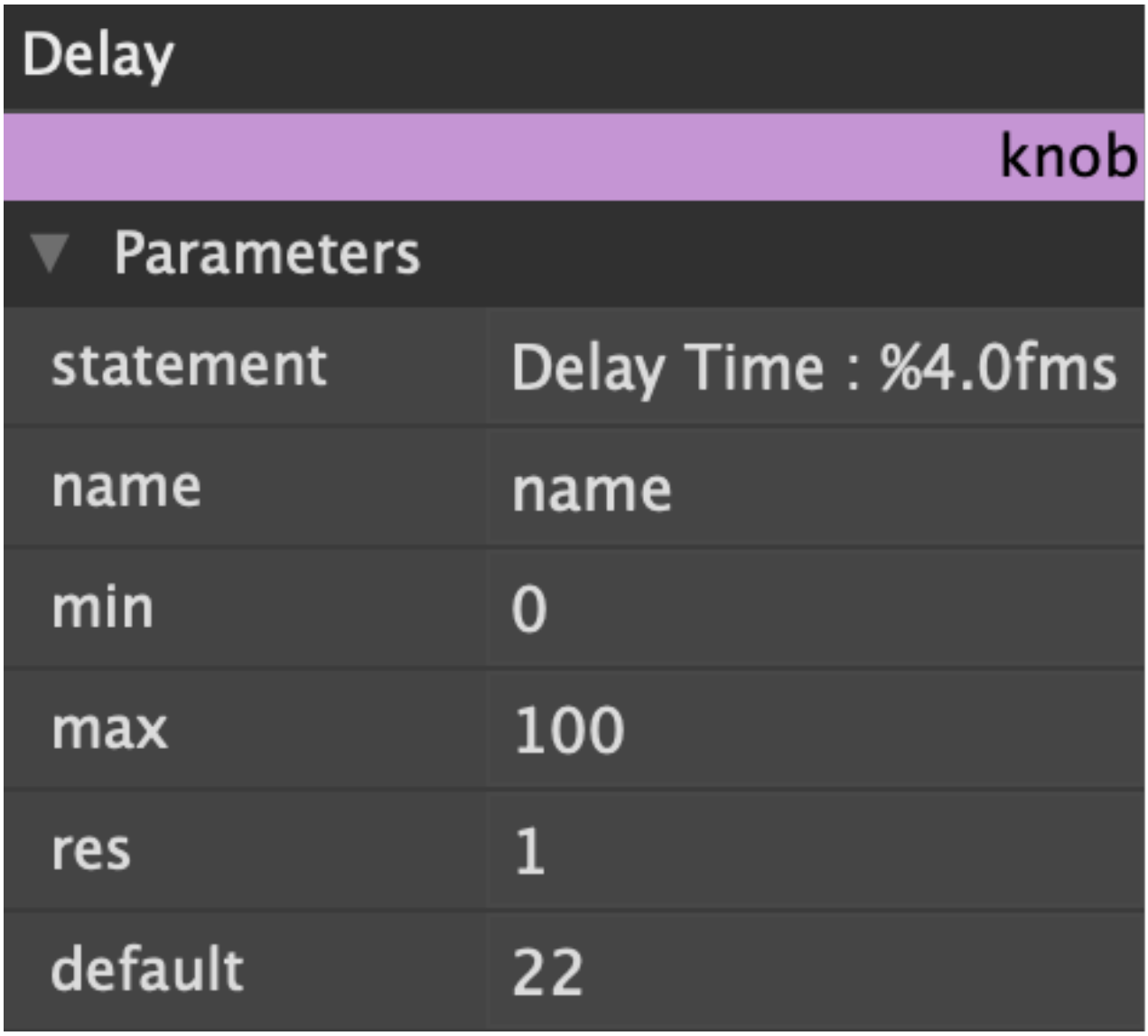


Fig. 6.23 **Parameter list for a *knob* module**

There are a few different parameters that can be adjusted. Their uses can be found in the documentation by right clicking the module then clicking "Show Module Documentation."

We can see that there are some values here.

- Statement - Text description in the PARAMETER menu
- Name – a short name which will be used if the knob is directly connected to the head
- Min - minimum value. Range -32768 to max
- Max - maximum value. Range min to 32768.0
- Res - multiplier of the output values, or step rate

- Default - initial value that is active when the algorithm is first loaded.

While most of these are self-explanatory, the statement parameter has a few tricks to best show the parameters.

To show the value of the variable associated, we can use the following syntax:

- %[flags][width].[precision]f is used for numerical values
- %[flags][width]s is used for text values
- Inserting a ':' will split the text into two lines, the first being below the knob, the second being above.
- To show a % symbol in the text, it must be typed using two % symbols: %%

Flags:

- '-' The value will be left aligned instead of right aligned
- '+' A '+'/'-' sign will always appear in front of the number
- '#' The displayed value will always have a decimal point

Example:

Fig. 6.24 **A knob module with a statement value 'DELAY TIME : %4.0fms'**

6.2.4.1. Menupages

After becoming comfortable with knobs and other simple control inputs, menus can start to be built. Working with the same chorus effect as before, we can add additional *knob* modules connected to the control inputs. However, to use multiple knobs, they must be connected to a *menupage* module. *Menupage* modules can be nested to make groups of interface modules. This allows your patch to be more easily navigable and more user friendly. We can see an example of this as follows:

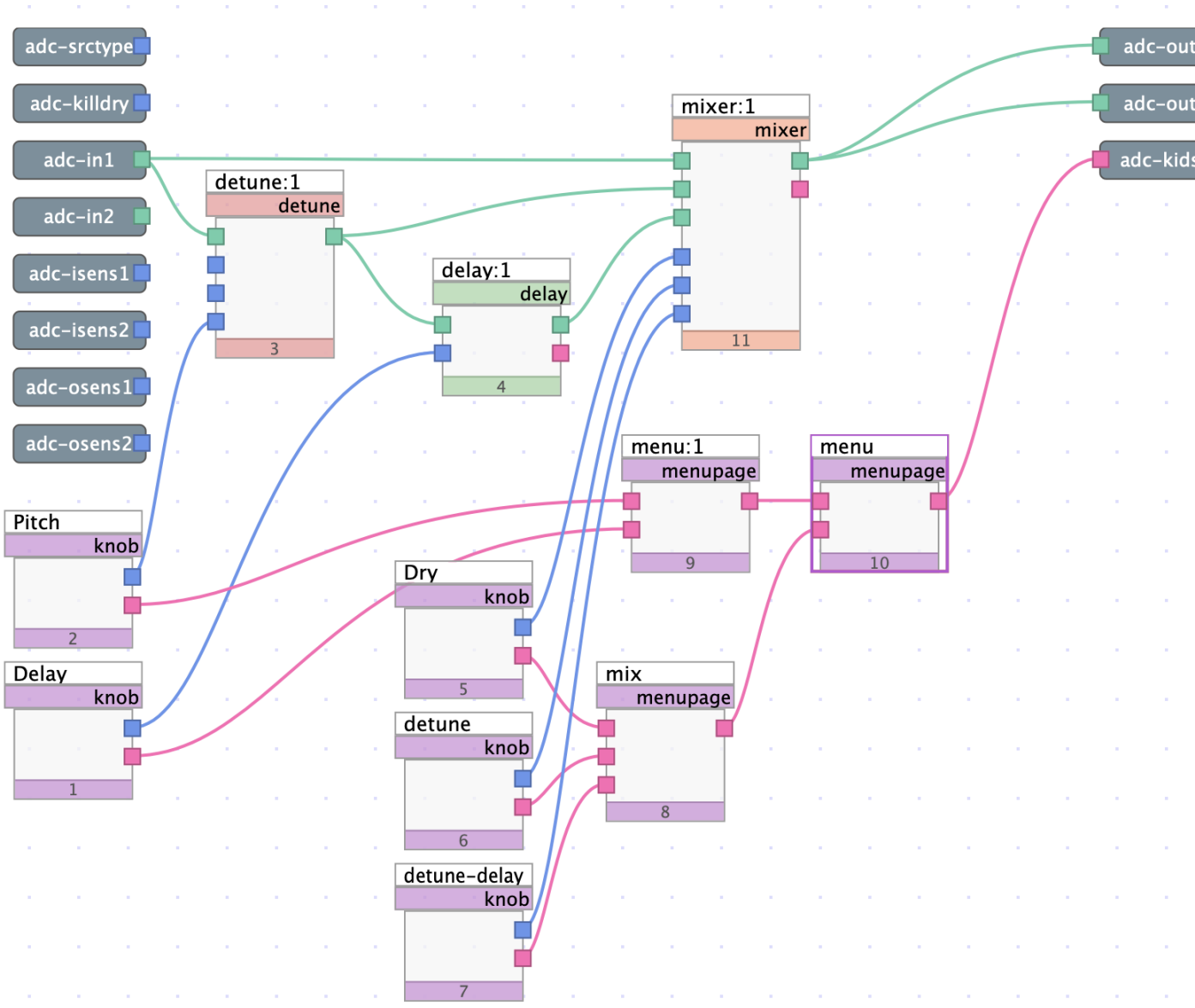


Fig. 6.25 **Simple chorus effect with several interface modules and menupages**

Fig. 6.26 **Menu displayed to user in Emote**

Download the sigfile here: Intermediate Menu Design1 and here: Intermediate Menu Design2 .

6.2.5. VSig Beginner Tutorial #5 – Graphic EQ

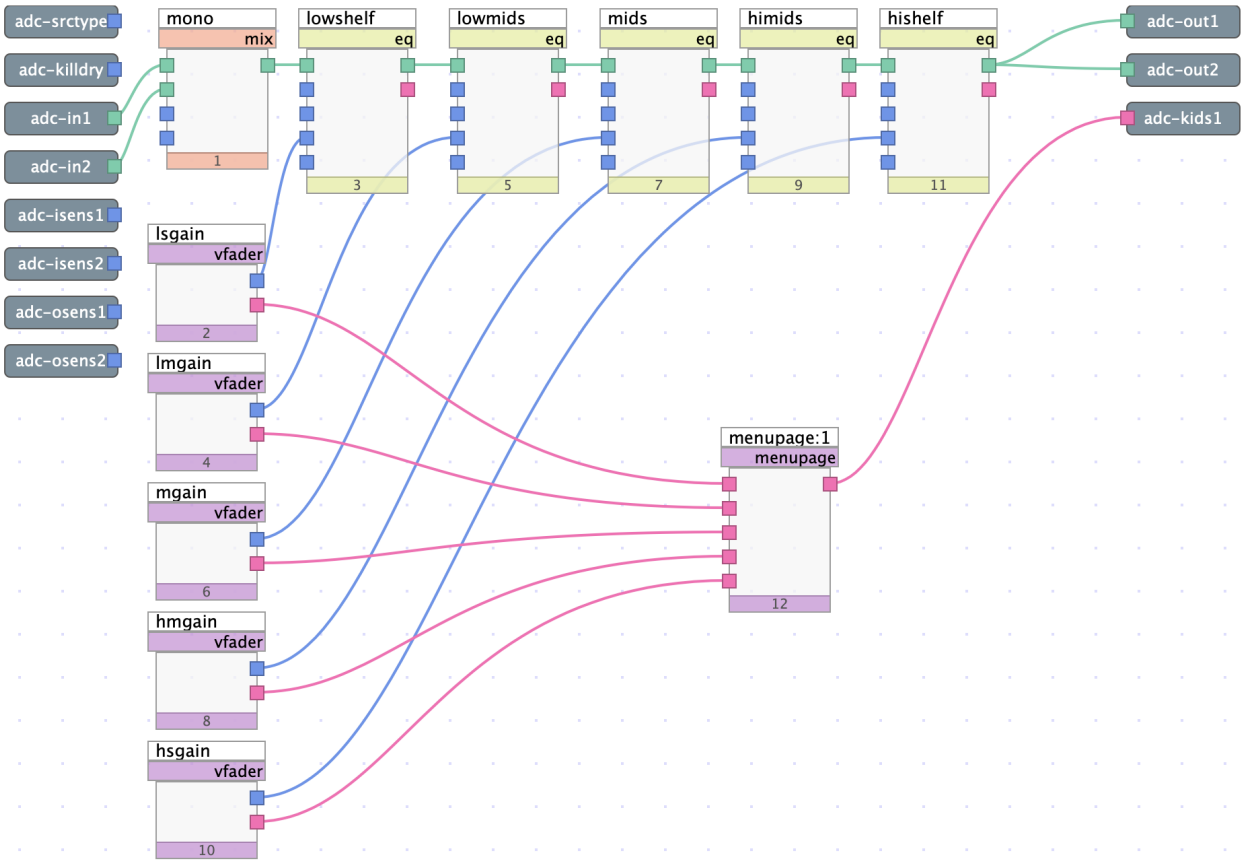


Fig. 6.27 **5-band graphic equalizer algorithm**

The algorithm above is a simple 5-band graphic equalizer. The center frequency and bandwidth of each band is predetermined, providing the user with control only over the gain for each band. The next beginner tutorial will expand on this algorithm, turning it into a parametric equalizer.

Download the sigfile here: Graphic EQ.

6.2.5.1. EQ Module pt.1

This algorithm uses five *EQ* modules in series, each of these modules representing a single band in our 5-band EQ. Let's take a closer look at the *EQ* module editor:

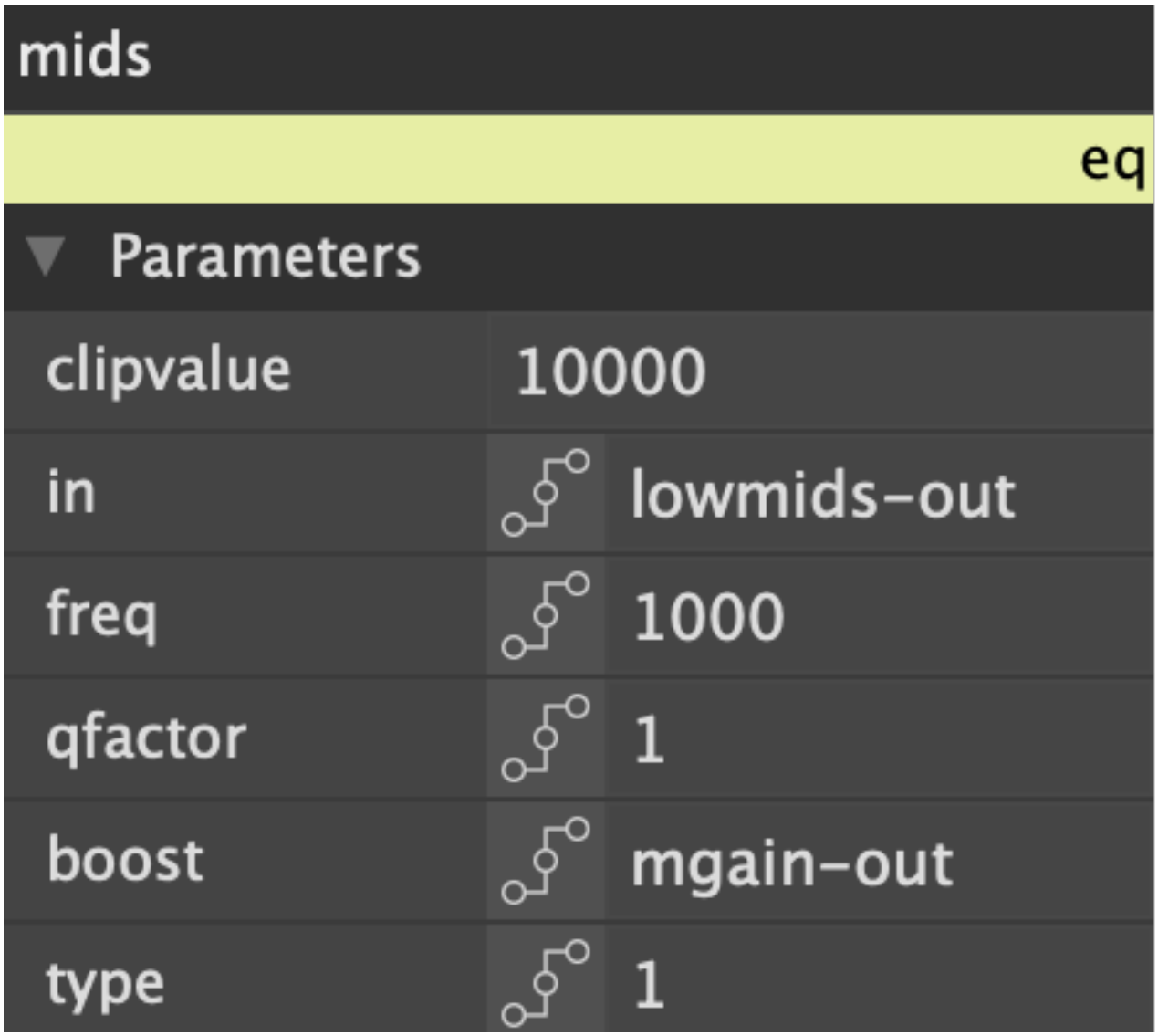


Fig. 6.28 **Module editor for *EQ* module**

Most parameters have been set manually, except for "boost" which is controlled by a fader. This specific *EQ*, designed to target the mid frequency range, has its center frequency manually set to 1000 Hz. Each *EQ's* center frequency, in ascending order, is 100 Hz, 330 Hz, 1000 Hz, 3300 Hz, and 10000 Hz.

The "type" parameter specifies whether the *EQ* is a low shelf (0), peak/notch (1), or high shelf (2). The *EQs* titled "lowmids," "mids," and "himids" all have 1s in their "type" fields. "Lowshelf" uses a 0 in its "type" field, while "hishelf" uses a 2.

6.2.5.2. Interface Module pt.1

The *vfader* (vertical fader) interface module is used to control the boost/attenuation of each EQ band. Each fader allows for up to 18 dB of boosting or attenuation for each band.

Upon exporting to Emote, the user is presented with five faders, each corresponding to a band in our 5-band graphic equalizer. Each fader is labeled with its center frequency, which cannot be changed by the user. Despite its simplicity, this algorithm is very easily upgradable to allow for user control of center frequency and Q factor.

6.2.6. VSig Beginner Tutorial #6 – Parametric EQ

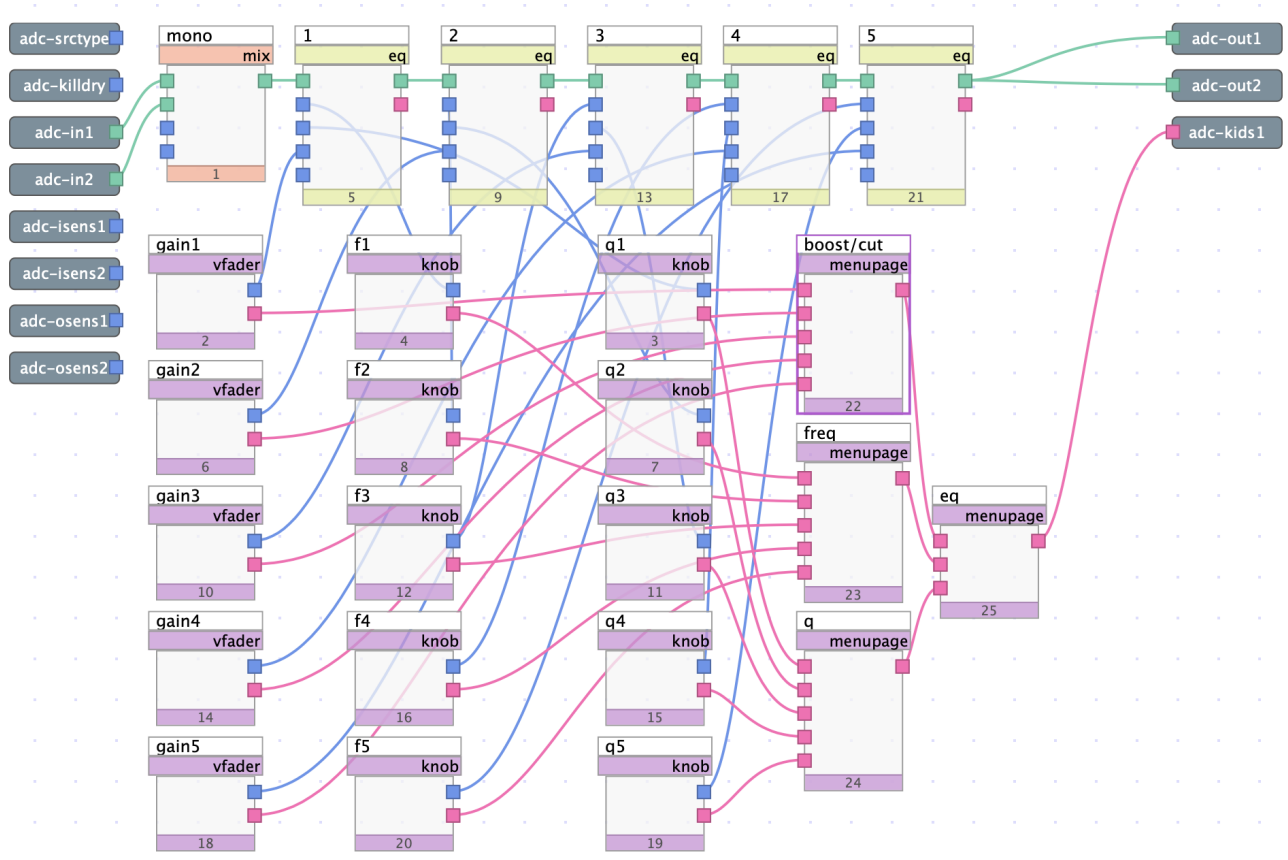


Fig. 6.29 **5-band parametric equalizer algorithm**

The algorithm above expands upon the previous tutorial, converting a simple graphic equalizer into a parametric equalizer with increased user control over EQ parameters. By just adding interface modules, a simple algorithm can become a much more powerful tool.

Download the sigfiles here: Parametric EQ1 and here Parametric EQ2.

6.2.6.1. EQ Module pt.2

This algorithm utilizes the same construction of five series connected *EQ* modules, but now almost all parameters are controllable by the user. Parameter control is best demonstrated using the module editor:

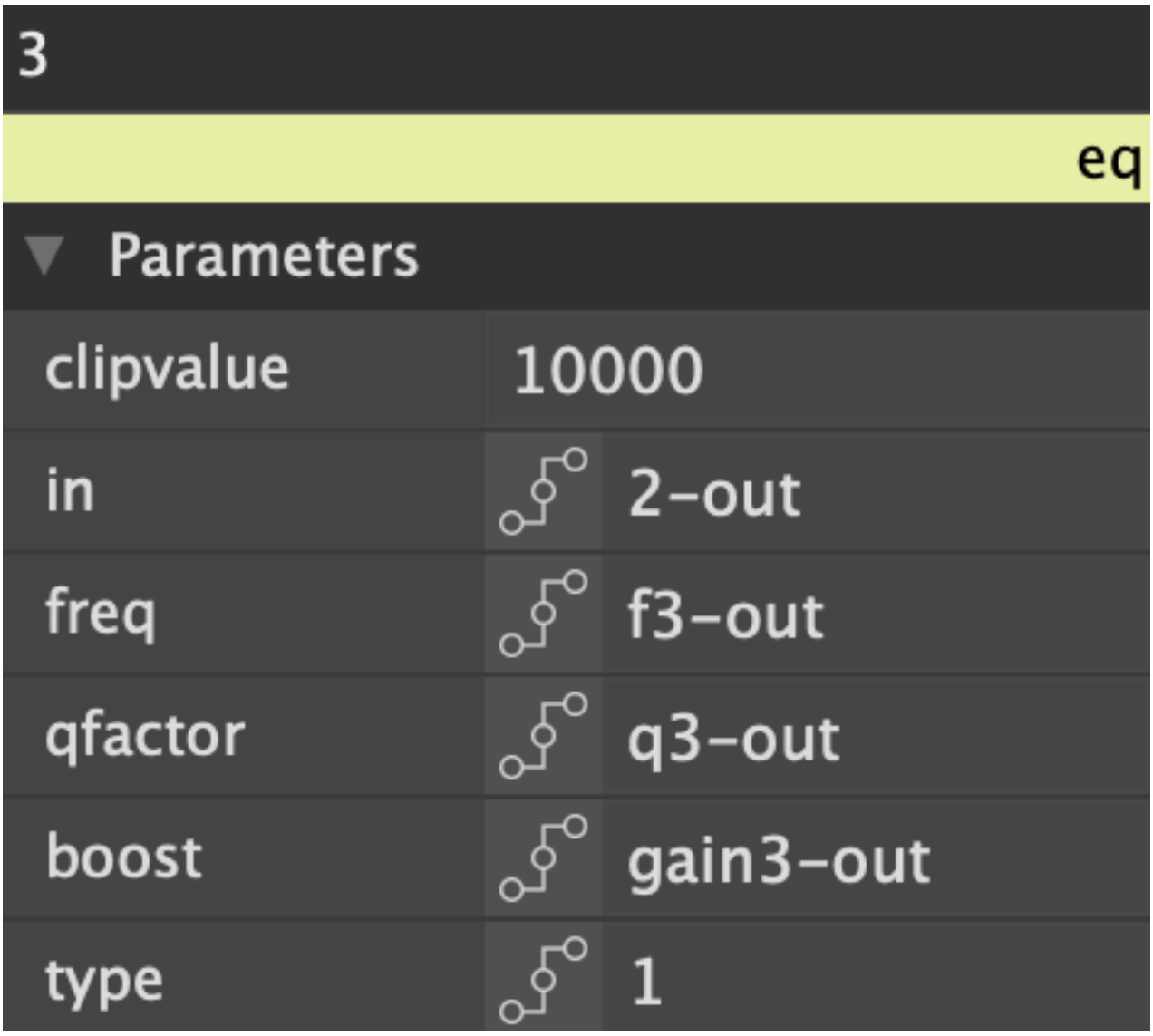


Fig. 6.30 **Module editor for *EQ* module**

Interface modules (*knobs*) now control the center frequency and Q factor of each EQ band in addition to boost/attenuation.

The “type” parameter remains defined by manual entry, so that the EQ is made up of a low and high shelf filter on both ends of the spectrum, with three peak/notch filters in between. For an even more customizable and modular EQ, the user may be given control over the type of each band (perhaps via a textknob).

6.2.6.2. Interface Module pt.2

Vfader continues to be used to control the boost/attenuation of each EQ band, while *knobs* are used to control both the center frequency and Q factor of each band. All Q factor *knobs* function identically, while each frequency *knob* varies in its "min" and "max" parameters.

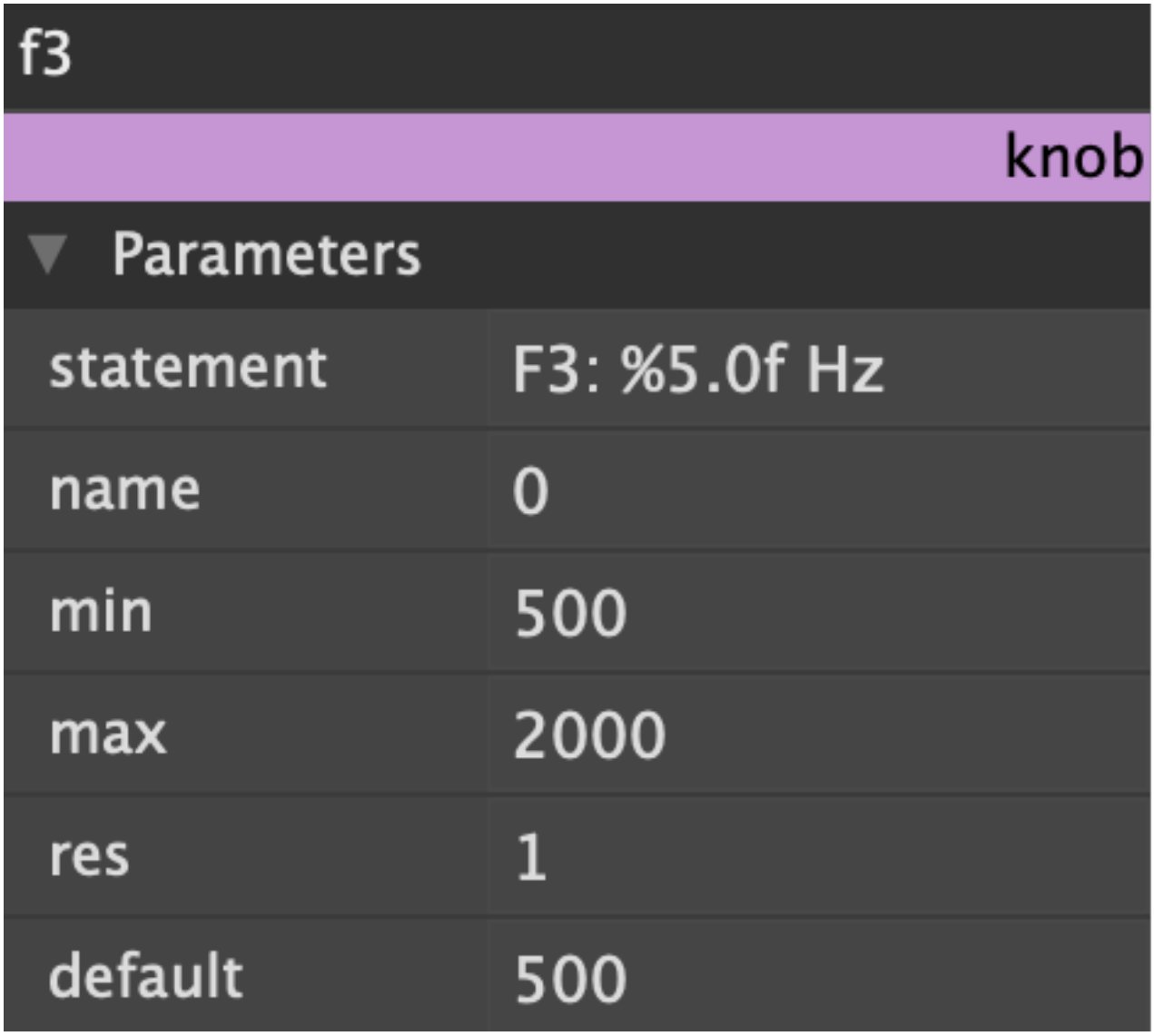


Fig. 6.31 **Module editor for *knob* module, which controls the third EQ band's frequency**

The module editor above shows the parameters of a *knob* which controls the center frequency of our third *EQ* module. The *knob* is limited to values between 500 and 2000 Hz, roughly encompassing the mid frequency range of the audible spectrum. In ascending order, the frequency ranges for each *knob* are:

- 20 Hz – 250 Hz
- 250 Hz – 500 Hz
- 500 Hz – 2000 Hz
- 2000 Hz – 4000 Hz
- 4000 Hz – 20000 Hz

Another solution for defining the frequency range of each knob could be to use *sknob3*, which accepts a control input for both its maximum and minimum values. Doing so would allow for each *knob* to have variable frequency range dependent on the output values of the other knobs. Integration of *sknob3* for frequency control is left as an exercise for the reader.

6.2.6.3. EQn Series Equalizer

Up to this point, several *EQ* modules have been used to create our EQ algorithms. This is a perfectly acceptable way of creating a multiband equalizer, but it may be easier (and more organized) to instead use one instance of the *EQn* module. *EQn* is composed of up to eight series connected EQs all within a single module. Converting our current algorithm to instead use *EQn* would look something like this:

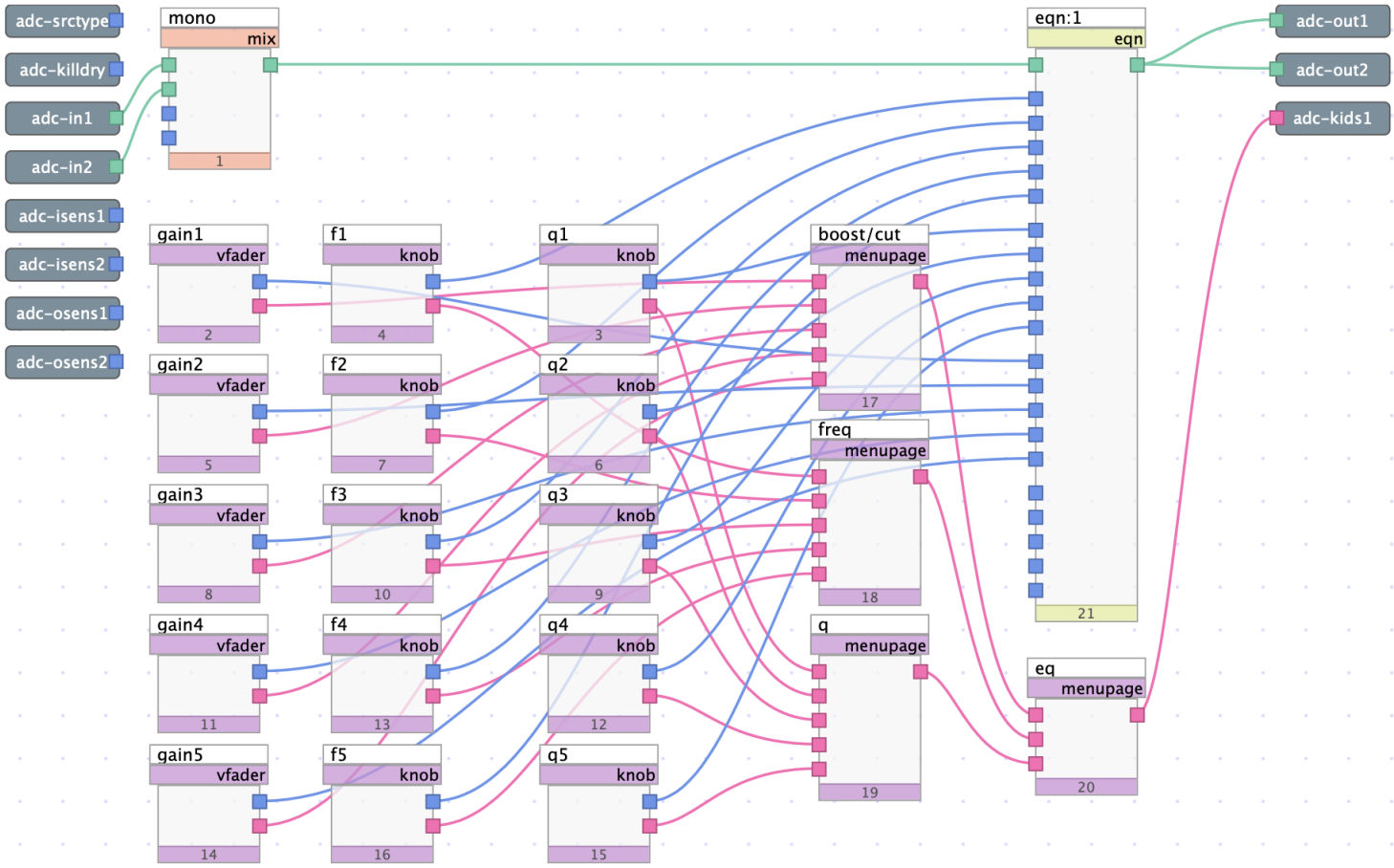


Fig. 6.32 **5-band parametric equalizer algorithm, using *EQn* instead of five *EQs***

All interface modules are completely unchanged, but now a single *EQn* module is used in place of 5 *EQ* modules. The resultant sound and usability are also unchanged, but the algorithm itself is slightly more concise.

Whether you decide to use several *EQs* or a single instance of *EQn*, the user will be presented with a set of five faders, and two sets of five knobs. Faders control boost/attenuation, one set of knobs controls the center frequency of each band, while the other set of knobs controls the Q factor for each band. While still quite a simple algorithm, the integration of expansive and intuitive user controls helps create a highly capable sound shaping tool.

6.2.7. Creating Reverb Algorithms in VSig

This short guide is meant to demystify VSig's reverb modules and show how they may be used with one another, or with other modules from other categories.

Download the sigfile here: Creating Reverb Algorithms in VSig.

6.2.7.1. "Plug and Play" Modules

6 of the available reverb modules can be considered *"plug and play"* in the sense that they do not require any additional modules to function as reverbs. **The highest quality reverb algorithms, however, will make use of other modules to better simulate acoustic space.**

Plug and play modules include:

- Reverb_2016
- Reverb_a
- Reverb_b
- Reverb_c
- Reverb_d
- Stereo_room

6.2.7.1.1. Reverb_a,b,c,d

Reverb_a,b,c, and d function identically and use the same parameters, the difference between them being delay density. The modules are of high, medium, low, and adjustable densities respectively.

Parameters

- Decay (s) - controls the decay time (overall RT60)
- Roomsize – adjusts the relative room size of the reverb
- Predelay (ms) - the amount of delay between the onset of sound and the onset of reverberant sound
- Low_freq (Hz) - controls the frequency at which LF attenuation works
- High_freq (Hz) - controls the frequency at which HF attenuation works
- Low_decay (dB) - controls the attenuation of frequencies below low_freq
- High_decay (dB) - controls the attenuation of frequencies above high_freq
- *Moddepth – controls the amount of delay randomization
- *Modrate – controls the rate of delay randomization

- *Modratespan – adjusts the degree to which different internal delays will be swept at different rates
- *Gliderate – adjusts the rate at which changes to delay values are “glided”
- **Delay(n) - controls the internal delays of the reverb; amount (n) of delays depends on which module is used (12, 8, 6, or selectable in multiples of 4)

*The reverb_n family includes built-in delay randomization which is controlled by the modulation/glide controls. This randomization helps reduce flutter/resonances and is often best utilized by setting a constant value inaccessible to the user. **Overly high moddepth or modrate values may introduce undesirable sounds of their own.**

**Adjusting internal delay times is the primary means by which a distinct room “character” can be created. Used in combination with roomsize, adjustment of delay times allows for the creation of distinct, varied reverb sounds within a single module.

Things to know about the reverb_a,b,c,d family:

- Smaller roomsize values tend to introduce more fluttering, with the smallest room sizes sounding very harsh and metallic. This can be partially remedied using the modulation settings, but not completely
- Larger roomsizes may also have audible flutters, but can be much more easily fixed using the modulation settings without introducing further artifacts
- In practice, the perceived room size depends both on the roomsize parameter and the individual delay values

6.2.7.2. Diffusors

Diffusor – creates a dense field of delay repeats that is typically used to create reverberator structures; essentially a chain of series-connected all pass filters

On its own, the *diffusor* module almost acts like a regular *delay* module with a feedback loop. The way in which delay builds up overtime, however, is a better approximation of how echo density builds up in a real space (during early reflections) than some of the plug and play reverbs can achieve. For plug and play modules without a built-in diffusion control (*reverb_a-d*), *diffusor* modules may be necessary to create a more realistic buildup of echo density.

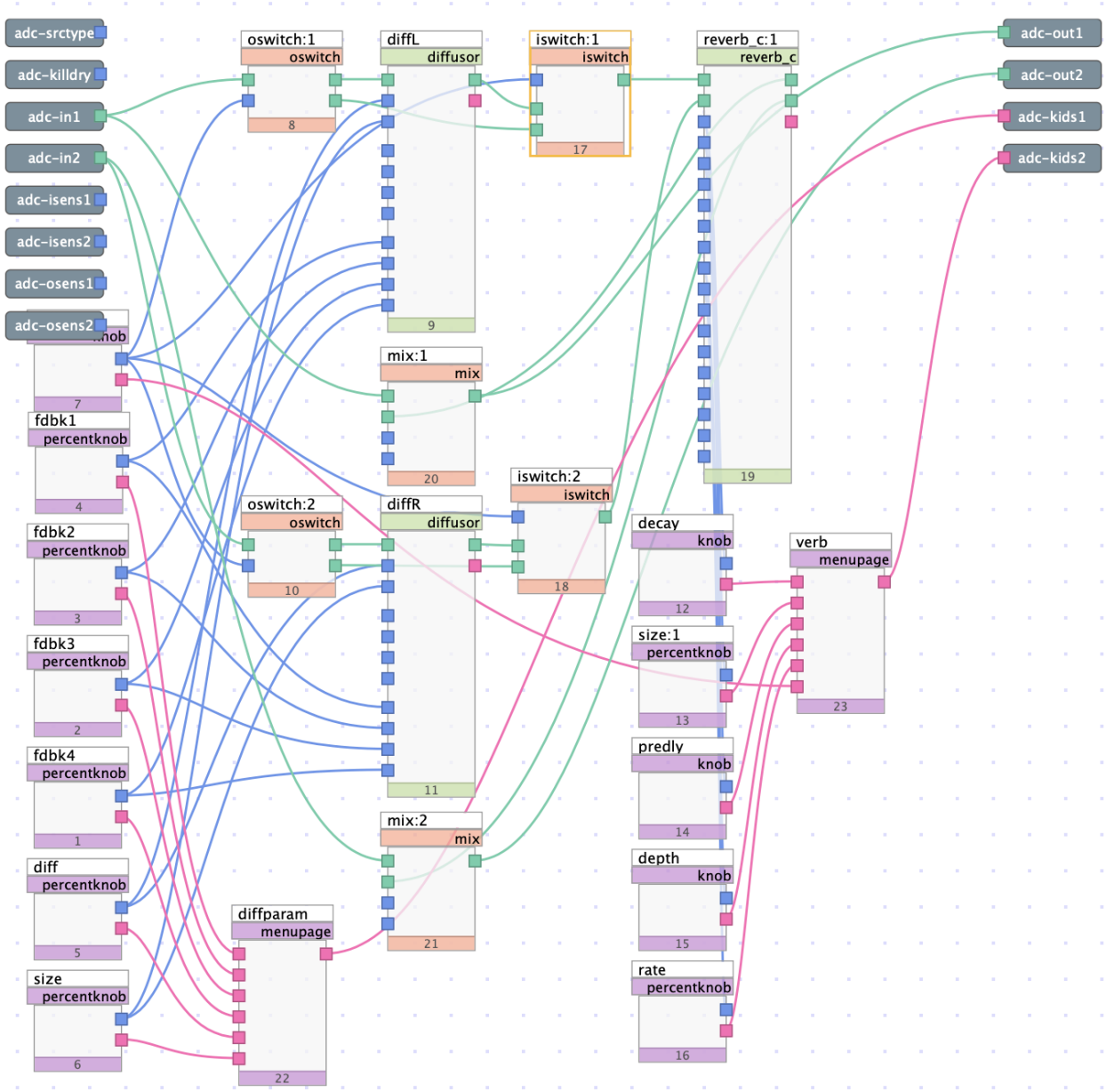


Fig. 6.33 **Reverb algorithm utilizing *diffusors* and the *reverb_c* module**

The above algorithm demonstrates how *diffusor* modules may be used in combination with *reverb_a-d* (*reverb_c* in this case). The presence of *diffusors* before the reverb module causes the reverb to swell more extremely and quickly than if signal was routed directly into the reverb module. To hear the difference for yourself, download the algorithm and toggle the diffusors between on and off using the *knob* labeled "Diff on." A value of 0 means the diffusors are on, and a value of 1 means they are off.

6.2.8. Creative Uses of Bridge Modules

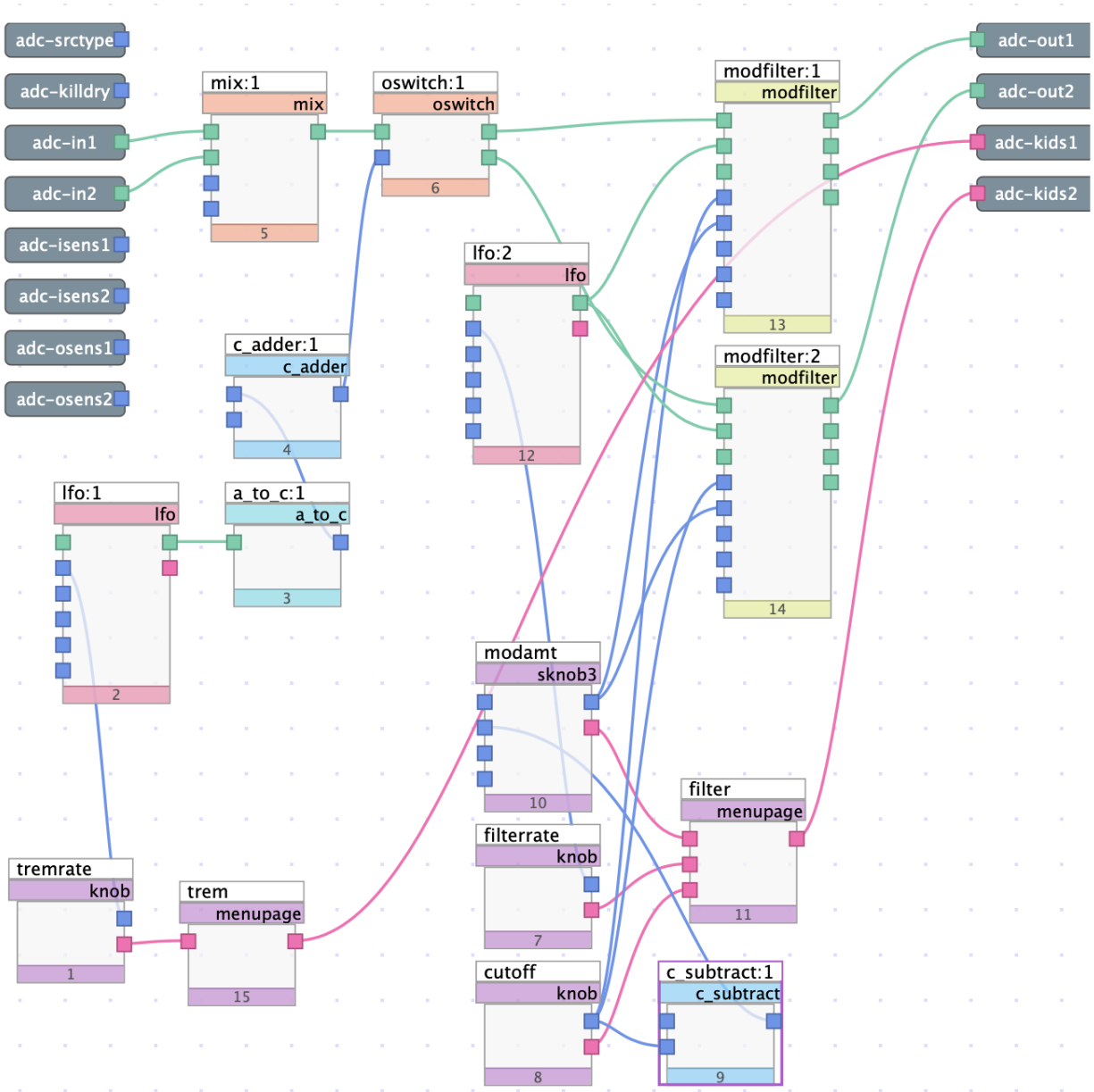


Fig. 6.34 **"Stereo Tremolo" esque algorithm using the *a_to_c* module**

The above patch demonstrates how bridge modules, in this case the *a_to_c* module, can be used for creative effects. In this case, an *LFO* is used in combination with *c_adder* and *a_to_c* to control an *oswitch* which only outputs to the left or right channel at any given time. The result is a stereo tremolo effect, used in combination with modulated filters for a "phase-y" sound.

The resultant sound of this algorithm includes two distinct effects: a stereo tremolo, which uses an *LFO* converted to a control signal to constantly switch between outputting to only the left or right channels, and an effect which has elements of a phaser and an autowah, achieved using *modfilters*. This algorithm is mainly designed to demonstrate one creative usage of the *a_to_c* bridge module, and how it may be used with other modules to create dynamic, multidimensional effects.

Download the sigfile here: Creative Uses of Bridge Modules.

6.2.8.1. Audio to Control Structure

The *a_to_c* module takes an audio signal input and converts it to a control signal output. In this case, an *LFO* is the audio input and is converted to a control signal on output. The converted signal is used to control a switch, which routes a single audio input to one of two outputs depending on the control input.

Before proceeding, let's dive deeper into how *oswitch* works. As previously discussed, *oswitch* switches between a given output based on the input control value. Output 1 is triggered when the control value is less than 1, and output 2 is triggered when the control value is greater than or equal to 1.

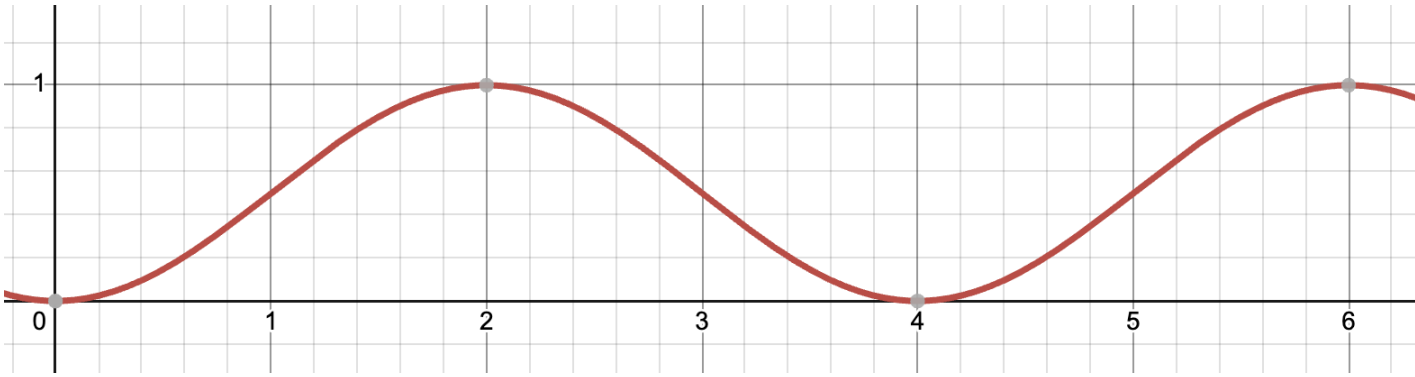


Fig. 6.35 **Time domain representation of the LFO output**

In this case, the control signal is constantly oscillating. By default, the *LFO* module (in unipolar mode) outputs in a range between 0 and 1. Clearly this control signal will never cause the *oswitch* to activate to output 2, as a value of at least 1 is required to do so. As a result, it is necessary to "vertically shift" our LFO signal in order to achieve the desired effect.

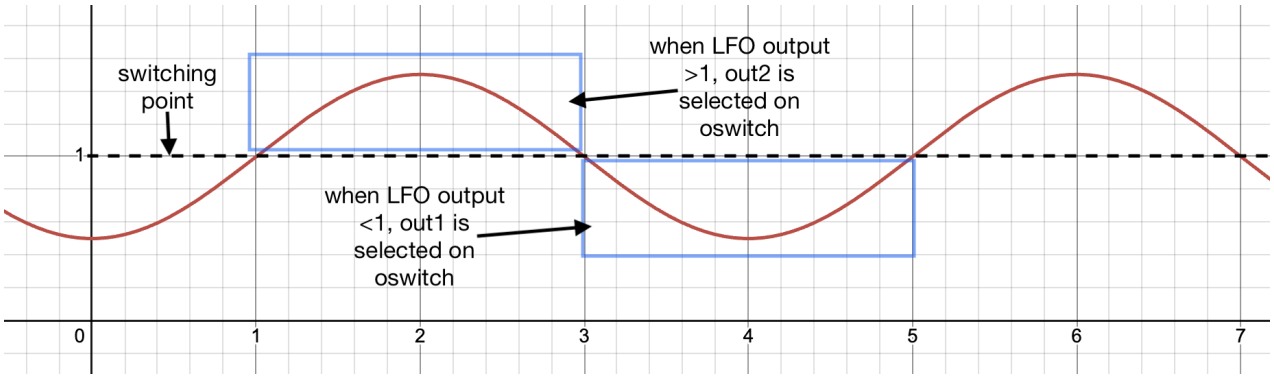


Fig. 6.36 **"Vertically shifted" LFO output, using *c_adder* module**

Vertical shifting is achieved using the *c_adder* module, which allows control signals to be added together. By adding a constant 0.5 to the *LFO* output, the signal outputs between 0.5 and 1.5. The result is a "ping pong" type effect, where audio constantly and uniformly switches between outputting to only the left or right channel.

6.2.8.2. Modulated Filters

Each switch output is routed to the input of a *modfilter* module. The cutoff frequency of each module is modulated by another *LFO*. The user is given control of the cutoff frequency, amount of frequency modulation, and LFO rate.

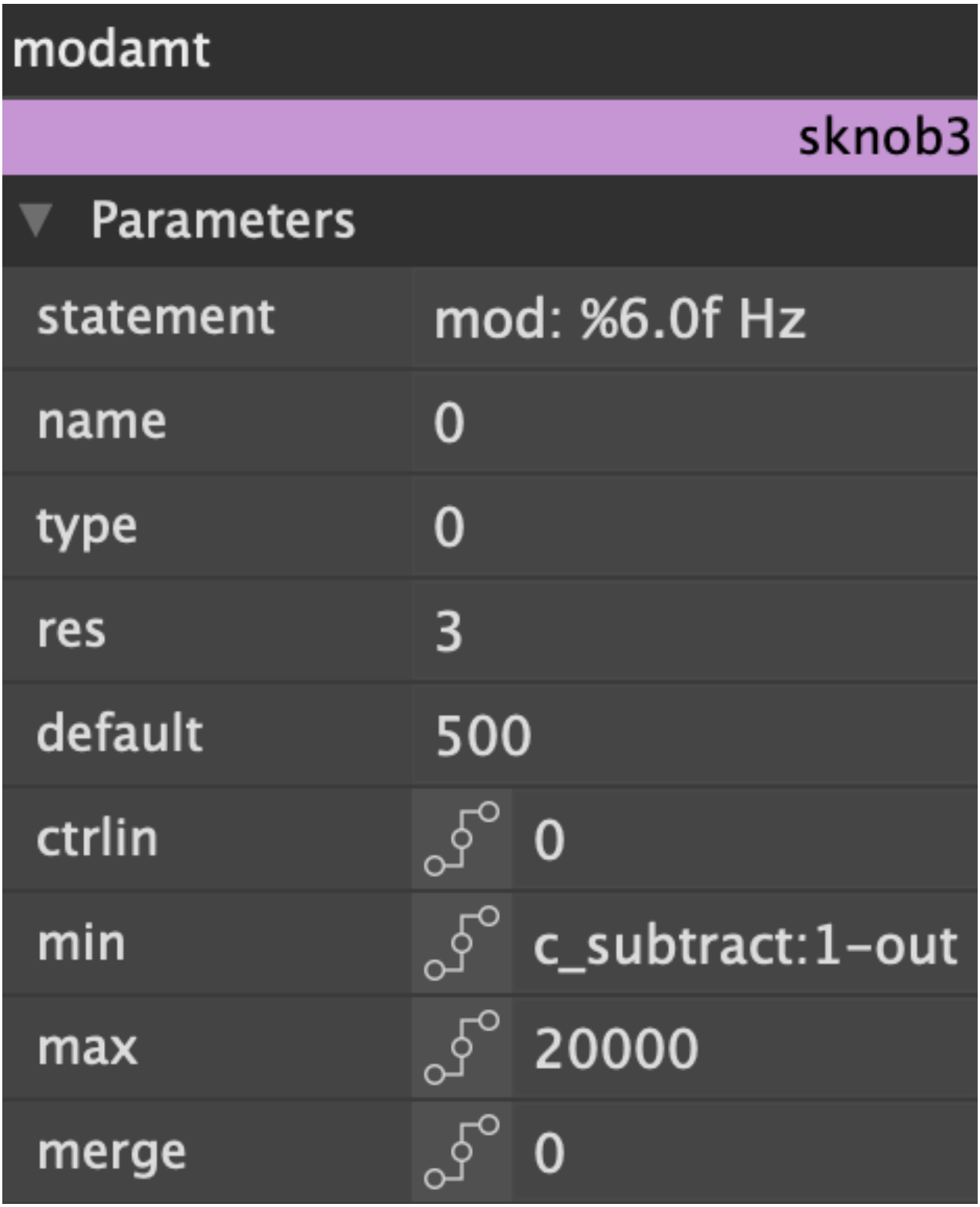


Fig. 6.37 **Module editor for *sknob3***

The *sknob3* module, used to control the amount of cutoff frequency modulation input, is used so that the minimum amount of frequency modulation is constantly dependent on the cutoff frequency. For example, with a cutoff frequency of 1000Hz, the user should only be able to modulate the cutoff frequency by as little as −1000Hz. This is achieved by using the control output of the cutoff *knob* to control the minimum possible value accessible by the modamt knob. The sknob3 module is one of the only interface objects whose minimum and maximum values can be affected by a control signal and is ideal for situations such as these.

6.2.9. Ducking and Sidechain Compression

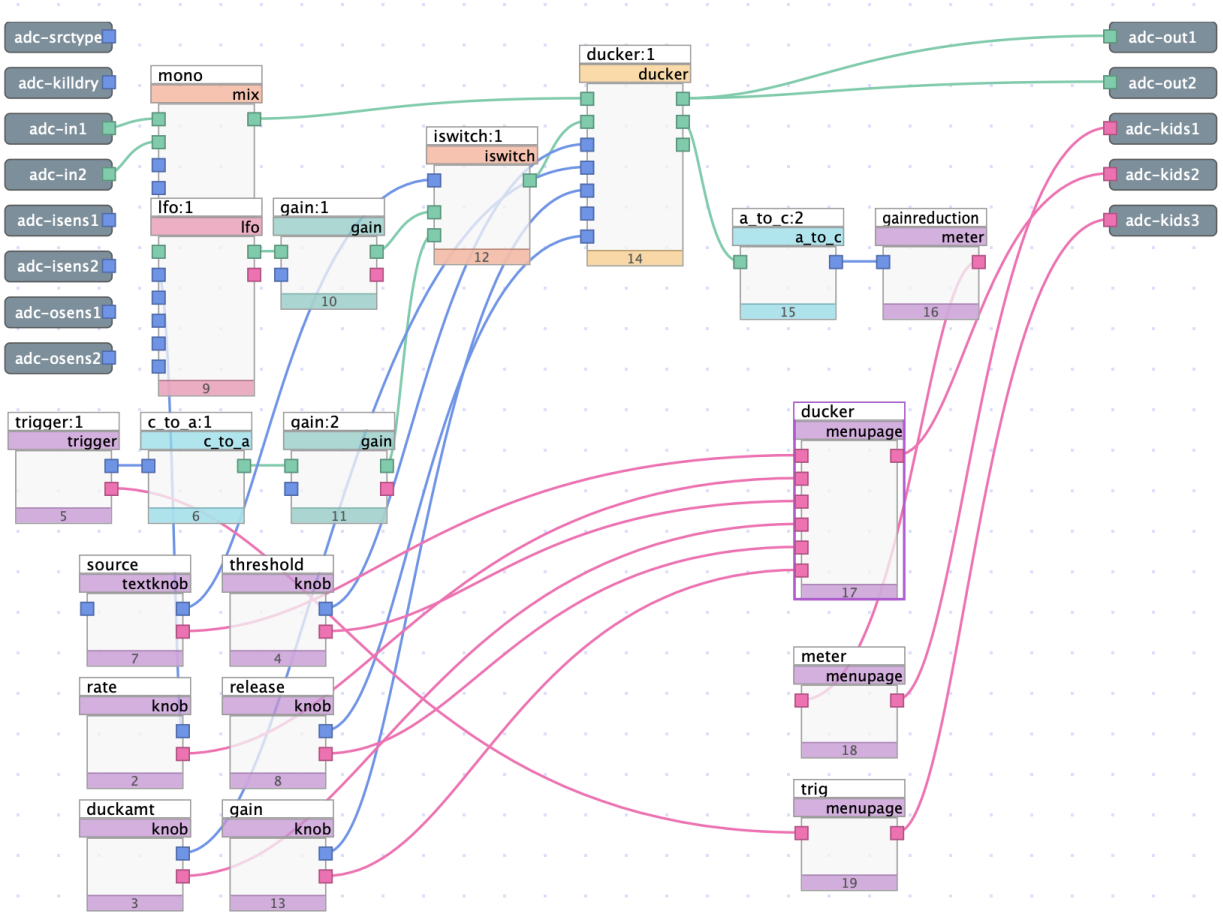


Fig. 6.38 **Automatic/manual ducker algorithm**

This algorithm explores a creative use of sidechain compression using the *ducker* module, which acts as a compressor with a sidechain input. By targeting the sidechain input with various audio sources, one may achieve a rhythmic, pulsating ducking, manually triggered ducking, and much more.

This algorithm acts as a ducker which can be set to duck either automatically or manually using the source toggle switch. In automatic mode, ducking occurs periodically at a user definable rate. In manual mode, ducking only occurs at the instant the “trigger” button is clicked. In both modes, the user has control over the threshold, amount of ducking, release time, and overall gain.

Download the sigfile here: Ducking and Sidechain Compression.

6.2.9.1. Sidechain with LFO

By default, this algorithm uses an *LFO* as its side chain input, specifically a ramp wave.

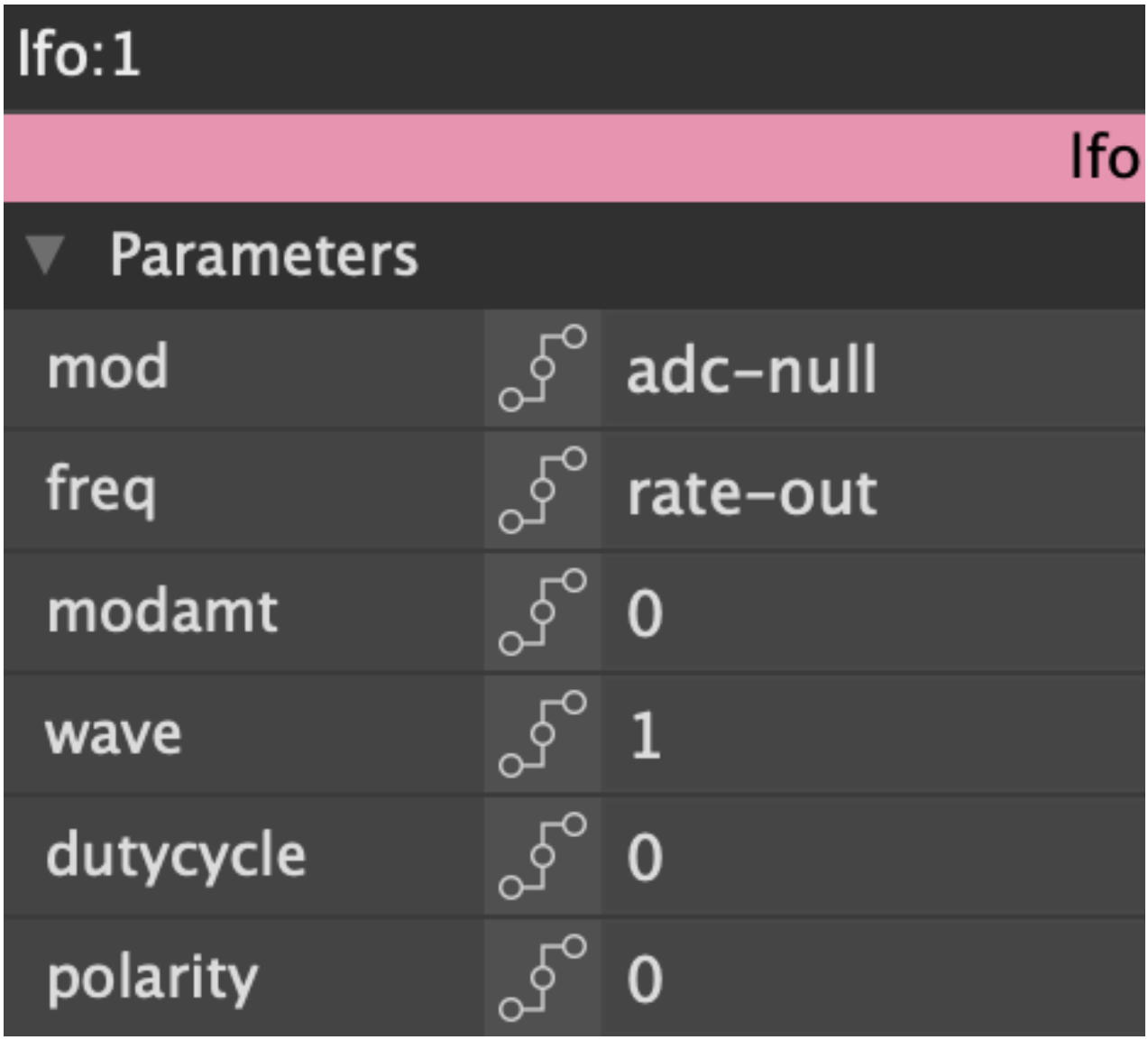


Fig. 6.39 ***LFO* module editor**

Looking into the *LFO* module documentation, you may notice that there is no option for a ramp wave shape. Instead, a triangle wave (indicated by a 1 in the wave field) with 0% duty cycle is used, effectively a negative ramp sawtooth wave.

Using an *LFO* causes gain reduction to occur periodically but at a uniform rate selectable by the user. A small amount of gain is applied to the *LFO* to ensure that it exceeds the threshold, although threshold and ratio can also be dialed in by the user to achieve a desired effect.

6.2.9.2. Manual Sidechain with Trigger

This algorithm also includes a source toggle in the form of a *textknob*. Toggling the switch changes the sidechain source to a *trigger*, meaning that gain reduction will only occur at the instant the trigger is clicked.

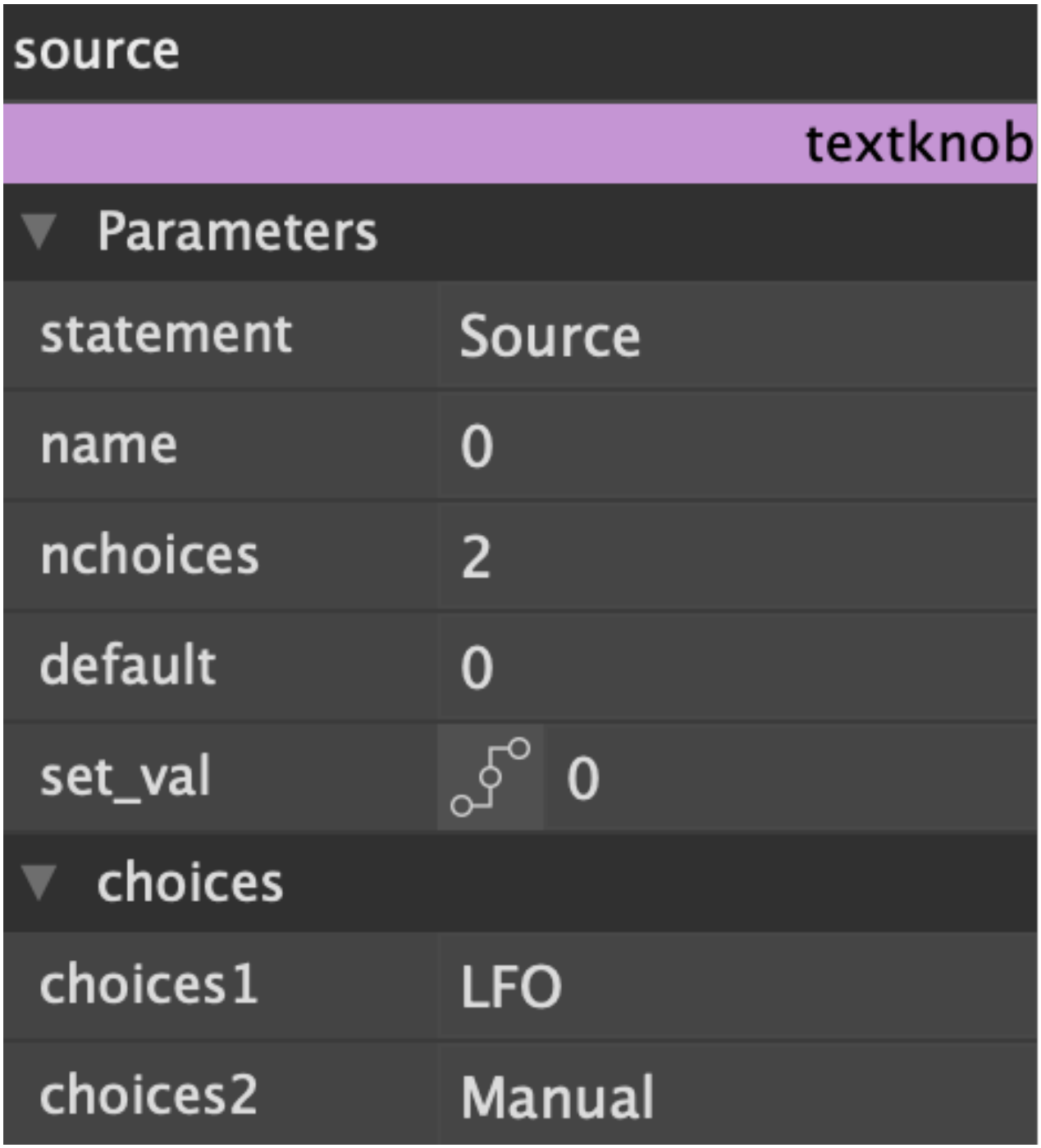


Fig. 6.40 ***Textknob*** **module editor.** ***Textknobs*** **with only two choices are automatically displayed as a binary switch, as opposed to a drop down menu. Despite both choices being labeled, these labels will not be shown to the user.**

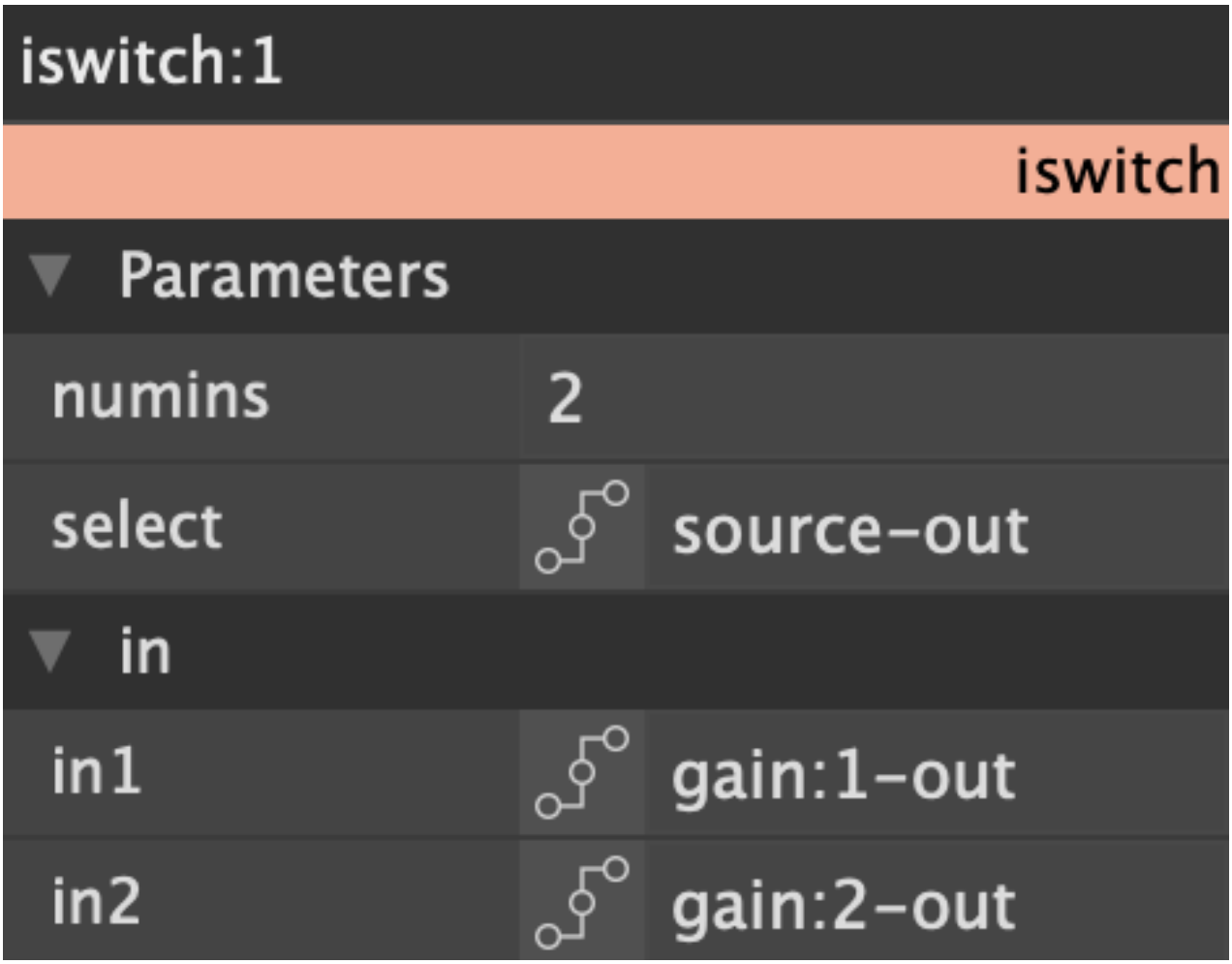


Fig. 6.41 ***Iswitch* module editor. The *textknob* shown above, labeled "source," acts as the control input.**

Because the *trigger* module outputs a control signal, a *c_to_a* bridge must be used so that the signal is accepted by the sidechain input. Gain is once more used to ensure that the trigger output exceeds the *ducker* threshold.

6.2.9.3. Metering

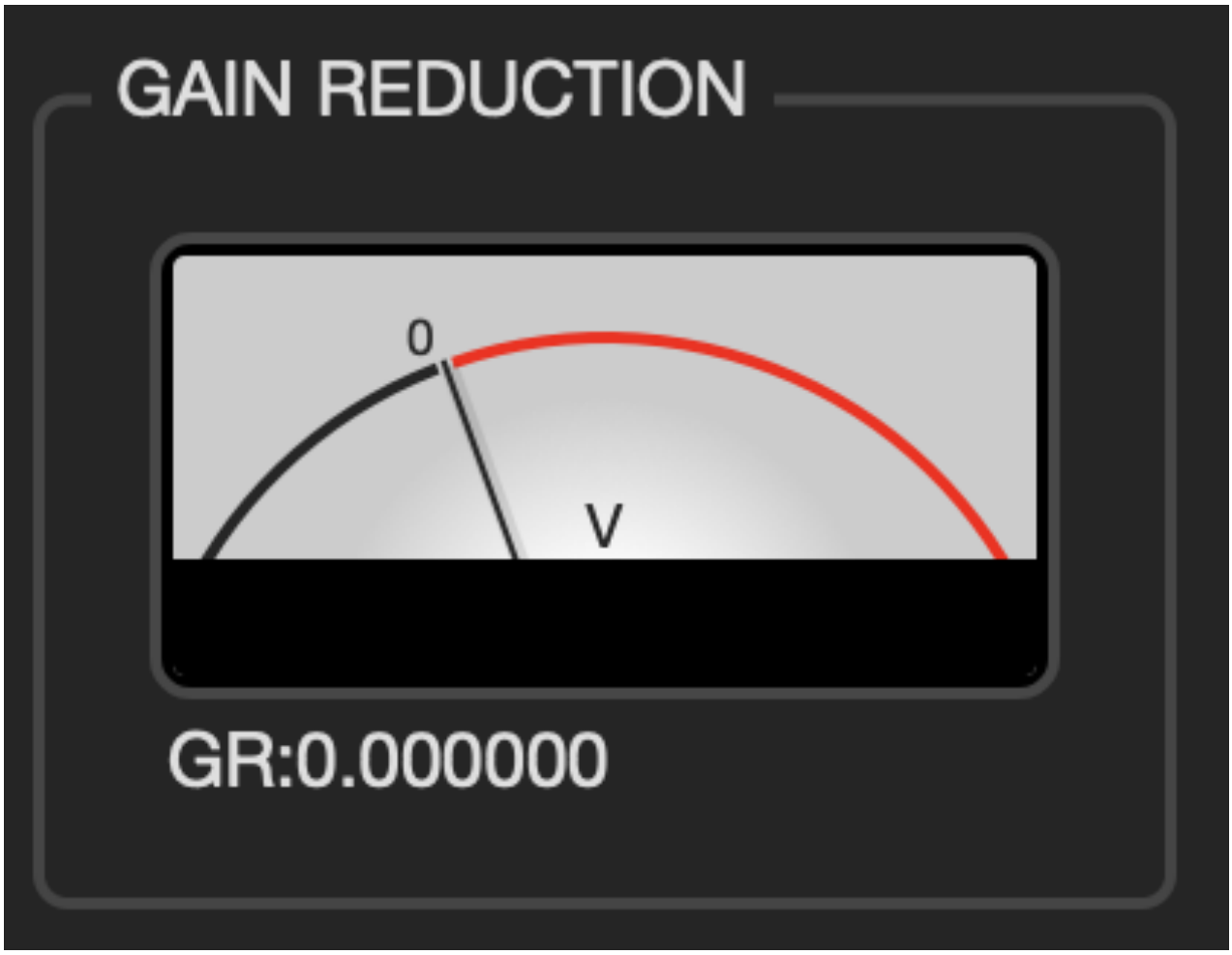


Fig. 6.42 **Analog style meter used to show gain reduction**

Users can ensure that the *ducker* is causing gain reduction using the *meter* module, which displays an analog style meter. An *a_to_c* bridge is required to display gain using the meter, as the gain output from the ducker is an audio signal.

6.2.10. MIDI Tutorial #1 – MIDI Tremolo

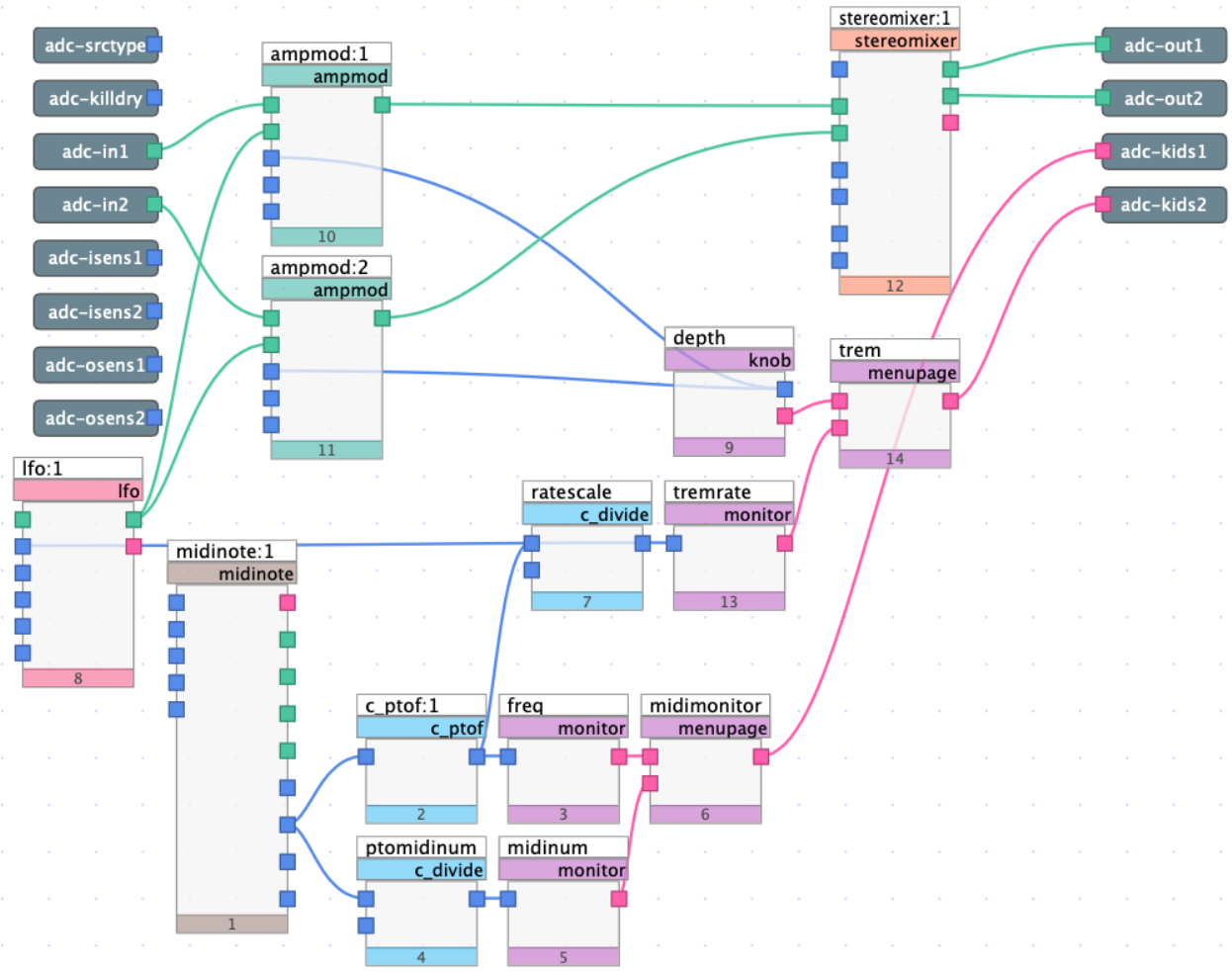


Fig. 6.43 **MIDI controlled tremolo effect**

Many of the algorithms included with the H9000 offer some sort of MIDI functionality. VSIG offers algorithm developers the ability to include MIDI functionality in their custom algorithms, using “External” type modules. The above algorithm uses *midinote*, a module which converts incoming MIDI messages to control signals, to control the rate of a simple tremolo effect.

In order to follow along with this tutorial, it is highly recommended to be using an external MIDI controller so that the algorithm’s main functionality can be accessed. Using external MIDI devices with the H9000 is as simple as connecting your controller to one of the USB type A ports on the front panel, or to the DIN port on the back panel. You can also send Midi to the H9000 from your DAW via the USB type B connection.

Download the sigfile here: MIDI Tremolo.

6.2.10.1. MIDI Monitoring

Before diving deep into MIDI, it is often helpful to set up a "MIDI monitor" menu so that one may view all data associated with an incoming MIDI message in real time, either in Emote or on the H9000 front panel. The VSIG patching required to create a MIDI monitor is as follows:

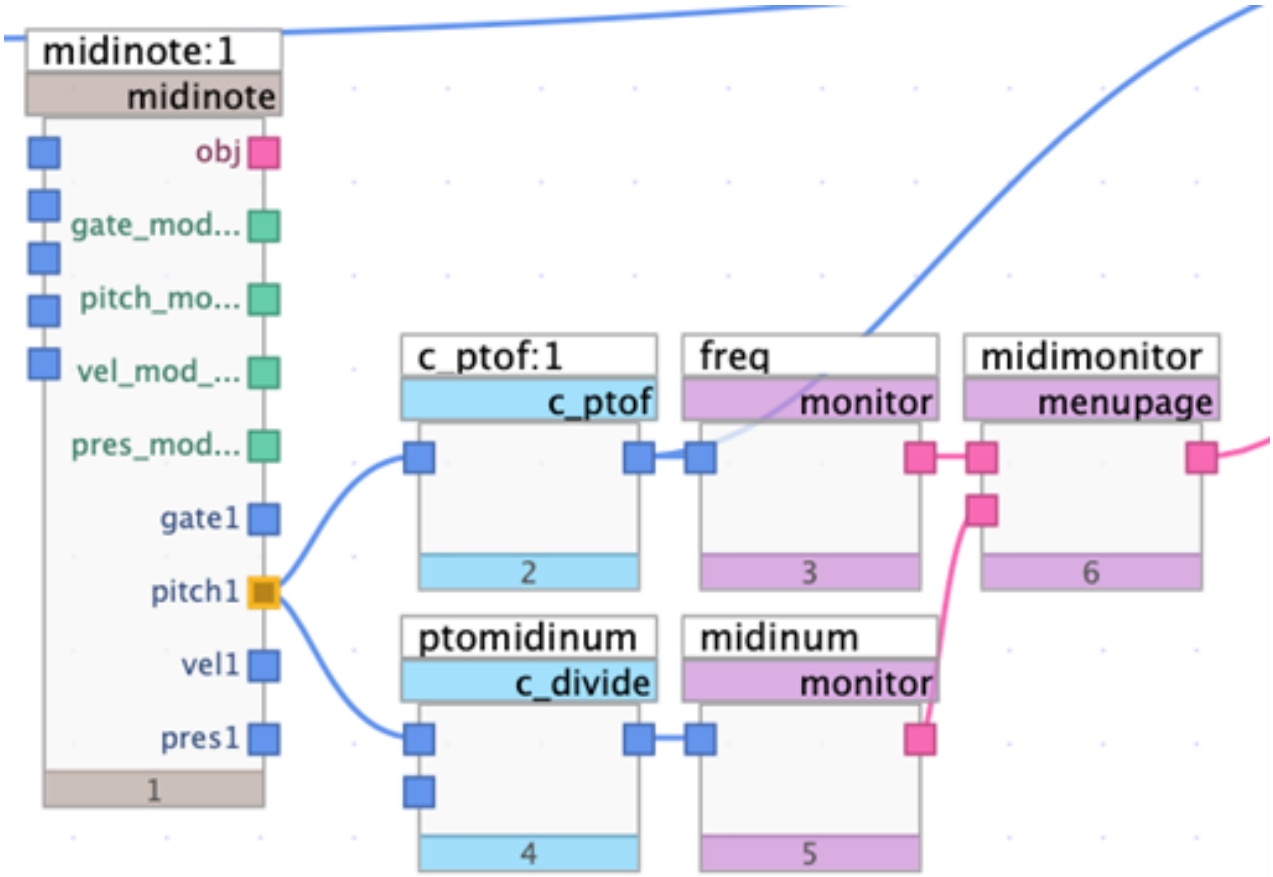


Fig. 6.44 **VSIG module structure used to display MIDI data to users**

This specific MIDI monitor is designed to display and MIDI number and frequency of an incoming MIDI note. Our *midinote* module outputs from its "pitch" control output, which outputs a pitch value in cents, to

two different controlmath modules, both of which are then sent to *monitor* modules. The controlmath modules convert a pitch value in cents to both frequency and MIDI number:

- Conversion from cents to frequency: VSIG makes this conversion quite simple with its *c_ptof* module, which converts from cents to frequency
- Conversion from cents to MIDI number: a MIDI note's MIDI number is simply its pitch, in cents, divided by 100 (*Side note: MIDI pitch is measured with reference to C0=0 cents*)

In Emote, the MIDI monitor looks like the following:

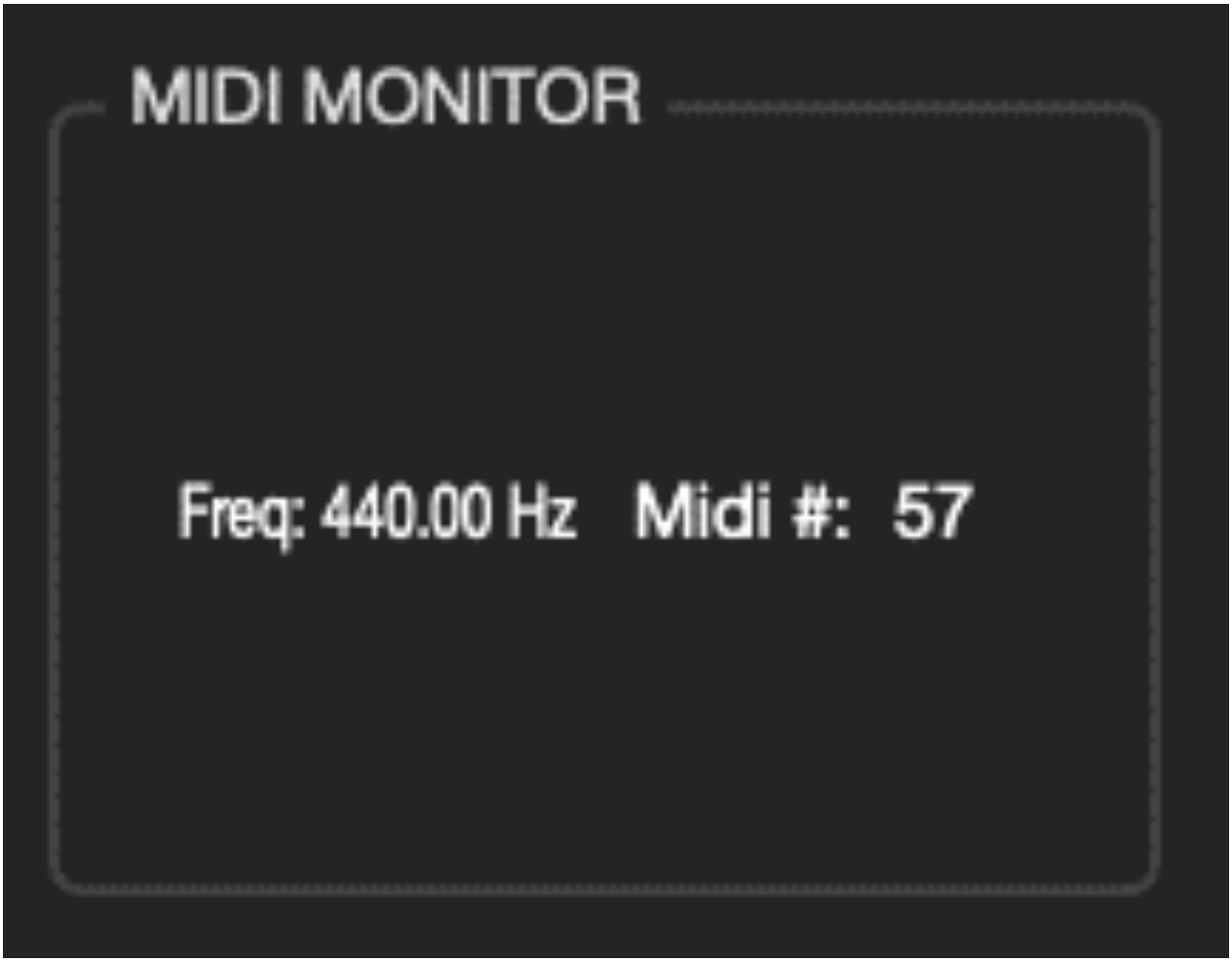


Fig. 6.45 **MIDI monitor in Emote, displaying the frequency and MIDI number of an incoming MIDI note**

Everytime a new MIDI message is received from an external MIDI device, the frequency and MIDI number update in real time. This is helpful when creating algorithms which utilize external MIDI control, providing developers with a real time view of incoming MIDI data.

6.2.10.2. Controlling Tremolo with MIDI

Now that the H9000 is receiving MIDI data that we can view in real time, it is time to start using MIDI to control other parameters within the patch. As previously mentioned, this patch is a MIDI controlled tremolo:

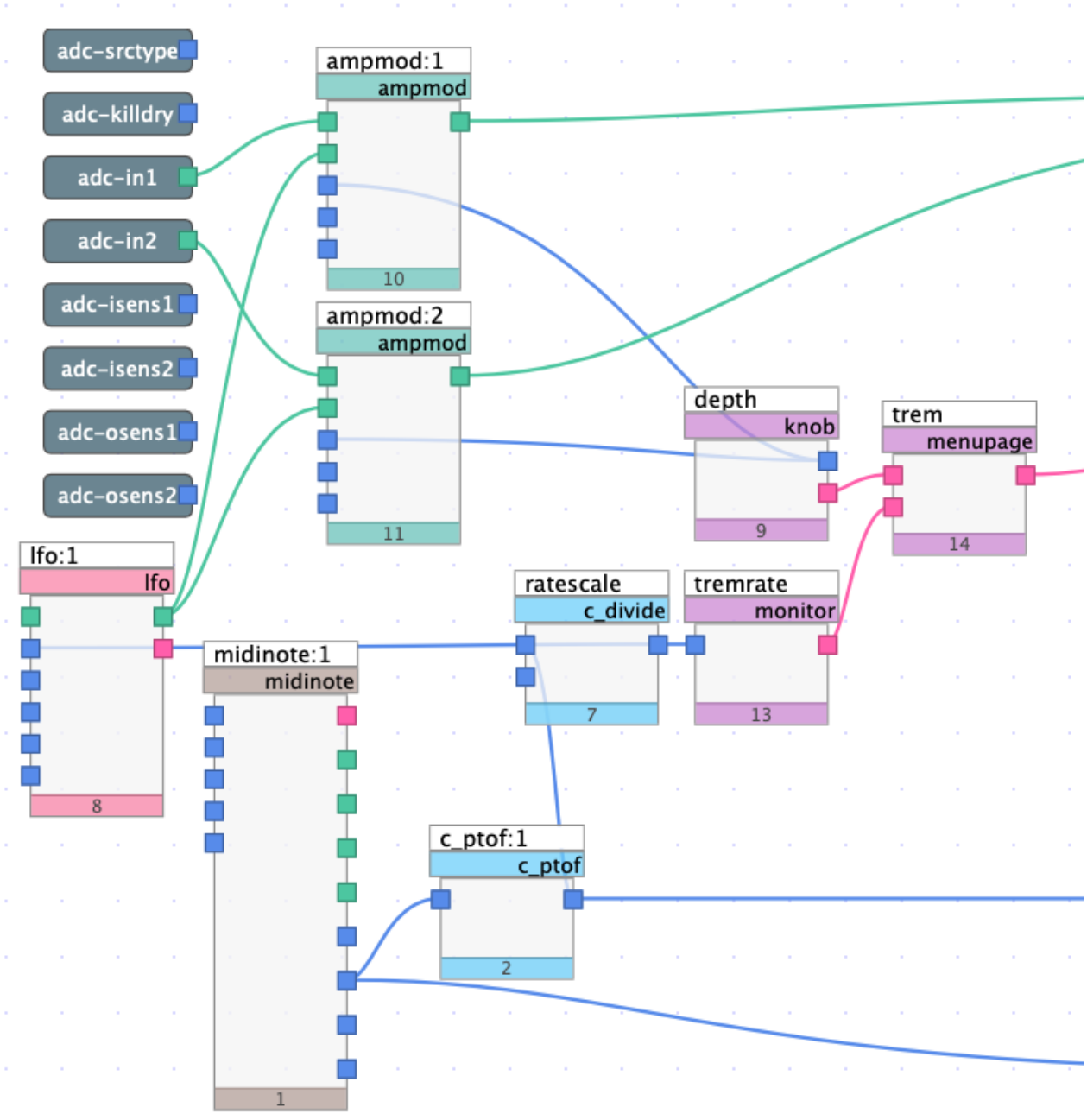


Fig. 6.46 **Isolated section of the VSIG algorithm showing tremolo structure**

The MIDI pitch output, converted to frequency, controls the rate of a *lfo* module. Subsequently, the *lfo* targets the mod inputs of two *ampmod* modules. In other words, the *lfo* modulates the amplitude level of an incoming signal, at a frequency defined by the incoming MIDI signal at a given time. Notice the *c_divide* module labeled "ratescale." This module divides the MIDI frequency output by 100, to restrict the rate of modulation from creeping too high (MIDI #127 is equivalent to a frequency greater than 20kHz, an absurdly high rate for a typical tremolo effect).

The amount of modulation is controlled by a simple knob, labeled "depth." In the next MIDI tutorial, we will explore how mod amount can also be controlled with MIDI.

Upon exporting to Emote, the following menus will be displayed:

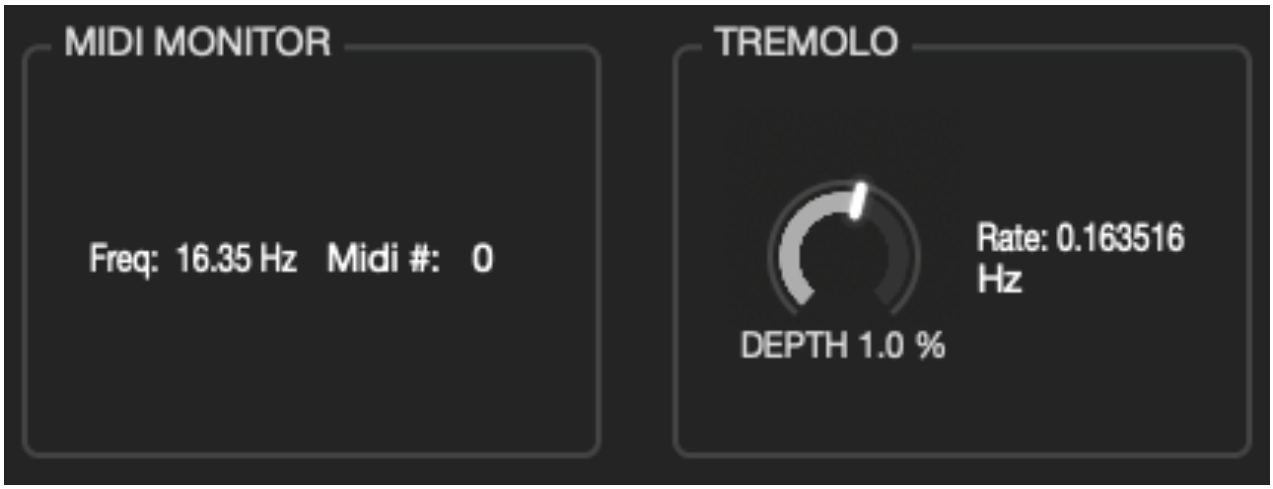


Fig. 6.47 **MIDI monitor and tremolo control menus, displayed to the user in Emote**

The "MIDI Monitor" menu displays data associated with incoming MIDI signals, while the "Tremolo" menu provides depth control and displays the tremolo rate. The rate of tremolo is controlled entirely via an external MIDI controller; the higher the note, the higher the rate. The following MIDI tutorial will upgrade the current MIDI tremolo by allowing depth to be controlled by key pressure.

6.2.11. MIDI Tutorial #2 – MIDI Tremolo V2

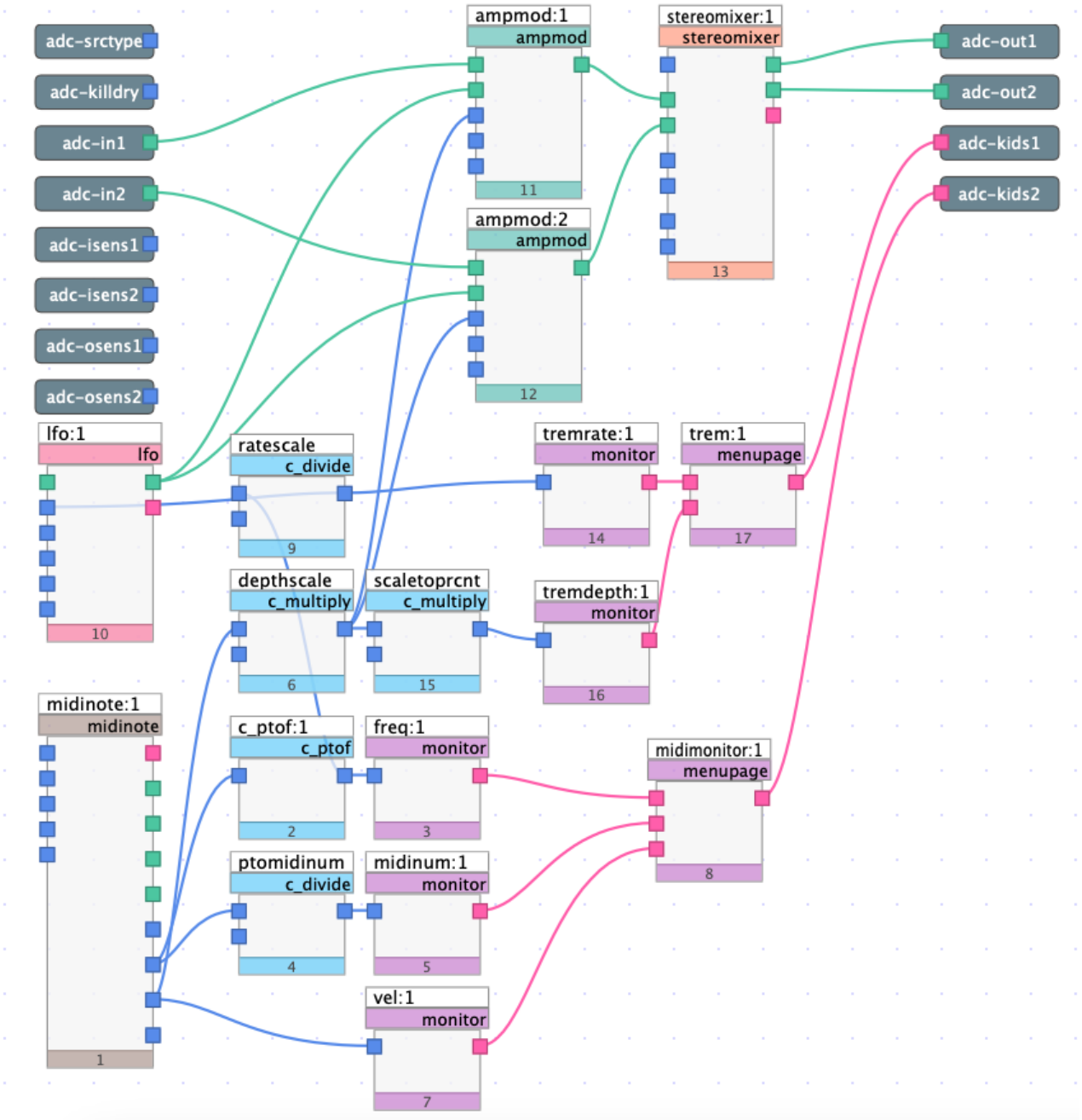


Fig. 6.48 **MIDI controlled tremolo effect, upgraded from previous tutorial**

The above algorithm is an expansion upon the algorithm explored in MIDI Tutorial #1. The amount of amplitude modulation is now dependent on MIDI key pressure, instead of being manually controlled by a

knob. This is achieved using the “velocity” control output from the *midinote* module to target the “modamt” control input of *ampmod*.

Download the sigfile here: MIDI Tremolo V2.

6.2.11.1. MIDI Velocity Monitoring

The previous MIDI tutorial explored creating a MIDI monitor menu to display data associated with incoming MIDI messages. This tutorial utilizes velocity, or how hard a key/pad is pressed on a MIDI controller, to control tremolo depth. Let’s add velocity to our MIDI monitor.

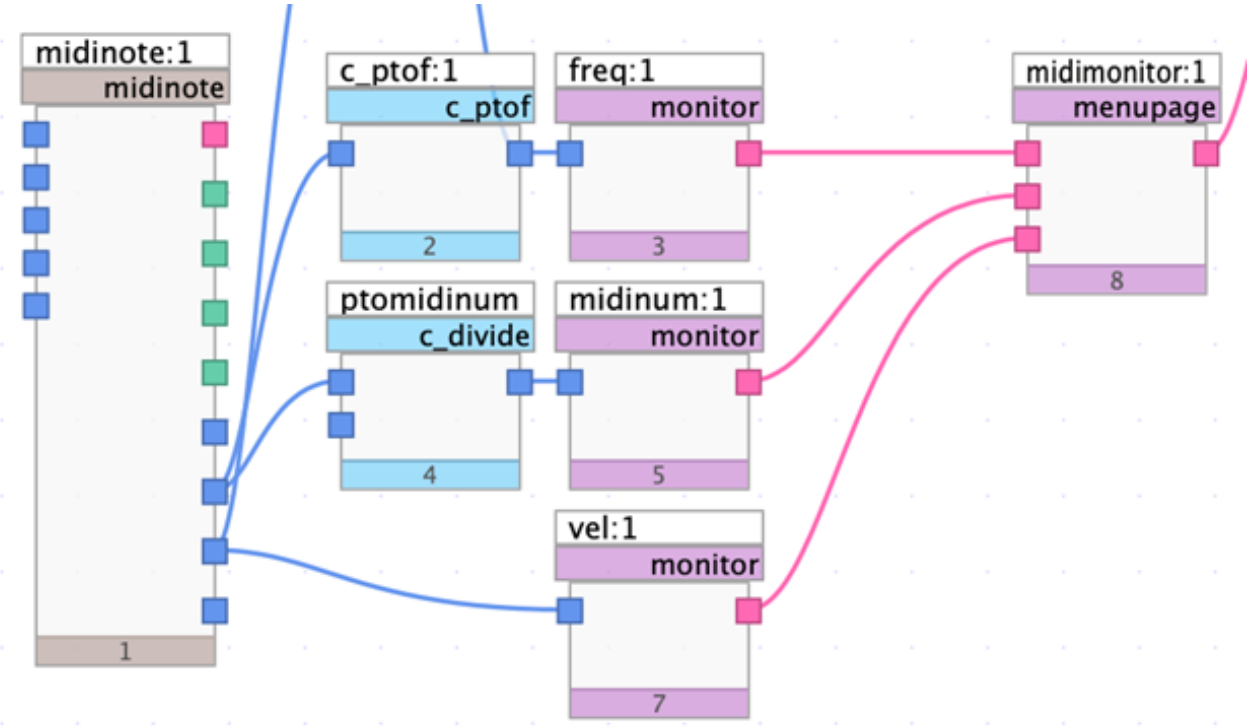


Fig. 6.49 **VSIG module structure used to display MIDI data to users**

Velocity does not require any control operations to be performed on it to be properly displayed to users. Midinote outputs velocity in a range of 0 to 1, 0 representing minimum key pressure and 1 representing maximum key pressure.

6.2.11.2. Controlling Tremolo Depth with Velocity

Now that we can view velocity in real time, the next step is to target *ampmod's* "modamt" control input with the incoming velocity value. The "modamt" parameter, however, accepts values from –10 to 10, whereas velocity only goes up to 1. We can remedy this by scaling velocity before routing it to ampmod.

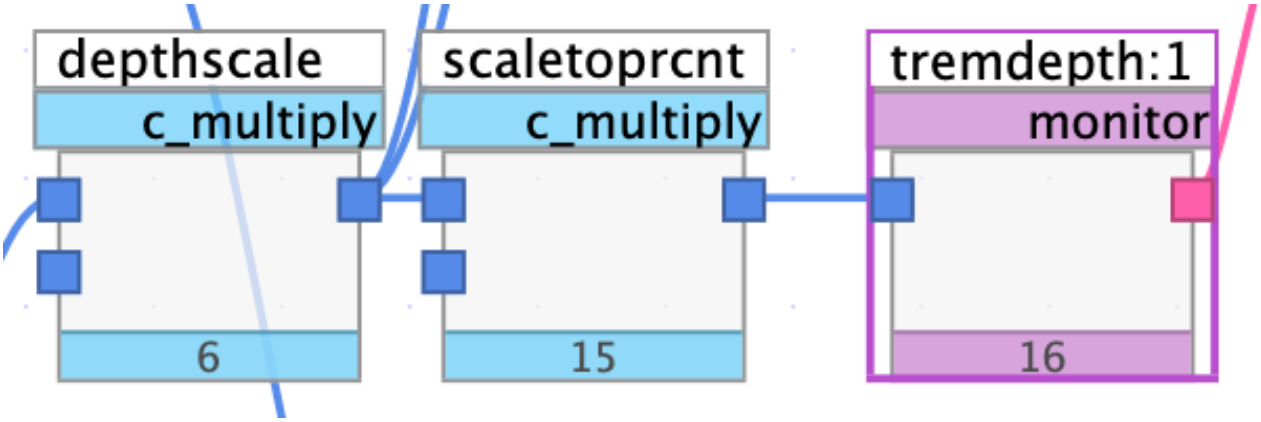


Fig. 6.50 **Scaling operations performed on velocity of incoming MIDI note**

Pictured above, the *c_multiply* module labeled "depthscale" multiplies the incoming velocity value by 10, after which it is sent to *ampmod's* "modamt" input. The second *c_multiply* module labeled "scaletoprcnt" multiplies the output of "depthscale" by 10 once more, but this value is only sent to a monitor. This is done so that depth may be viewed by users as a percentage, as opposed to an arbitrary value between 0 and 10.

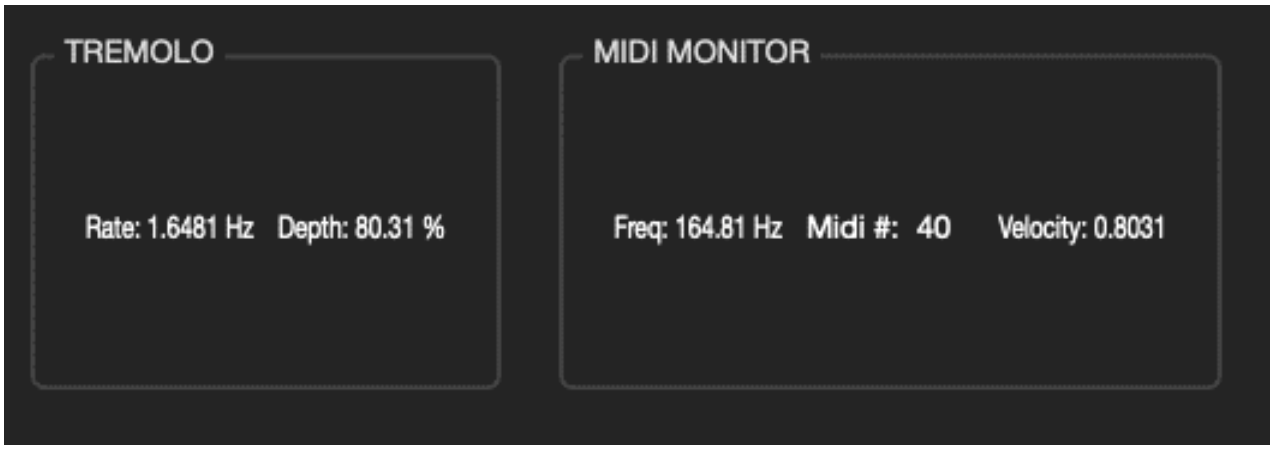


Fig. 6.51 **Menus displayed in Emote showing incoming MIDI data and tremolo parameters based off of MIDI data. Notice the relationship between "Velocity" and "Depth," as well as "Frequency" and "Rate."**

The output after these scaling operations is shown above. Notice how the displayed depth value is simply the velocity value multiplied by 100, and how the displayed rate value is the frequency value divided by 100.

Upon exporting to Emote, the menu pictured in the above section will be displayed to the user. While the algorithm features no controls in the UI, both tremolo rate and depth are entirely controllable via an external MIDI controller. Rate depends on the pitch of an incoming MIDI signal, while depth depends on velocity.

6.3. RNBO

RNBO for Max/MSP allows for the creation of Max-like patches which can be directly exported to targets such as audio plugins (VST/AU), source code (C++/JavaScript), or an external device like a Raspberry Pi.

RNBO also offers support for Eventide H9000, allowing users to upload their RNBO patches directly to the H9000 to be used as algorithms. The process is simple and similar to the process of uploading algorithms from VSIG to the H9000 but requires some set up.

System Requirements

- Full license of Max/RNBO
- Max version 8.6.0 or higher
- RNBO version v1.2.6-eventide.2
- Eventide H9000 package 1.0.3-beta.4
- H9000 firmware 2.1.12 or higher

6.3.1. Installation

To get the preview version of RNBO:

- Open Max's Package Manager
- Navigate to the RNBO page for "Remote Packages"
- Click on "Show All Available Versions"
- On the right side of this screen, find and select the entry *eventide* in the pop-up menu
- Choose the topmost entry in the list of distributions (there will only be one at the moment, *v1.2.6-eventide.2*)
- On the next screen, press the blue "Update to…" button and allow RNBO to install.

To get the preview version of the "Eventide H9000" package:

- Open Max's Package Manager
- Navigate to the "Eventide H9000" package page under "Remote Packages"
- Click on "Show All Available Versions"
- Select *beta* from the pop-up menu on the right.
- Choose (for now) *1.0.3-beta.4* and install it.
- Max should prompt you to restart, please do so: the RNBO target list is only updated at server startup (usually at application launch)

6.3.2. RNBO Setup

- Follow the installation instructions above.

- Open a RNBO patch, here we are using the RNBO FreqShifter example from the RNBO Guitar Pedals collection.

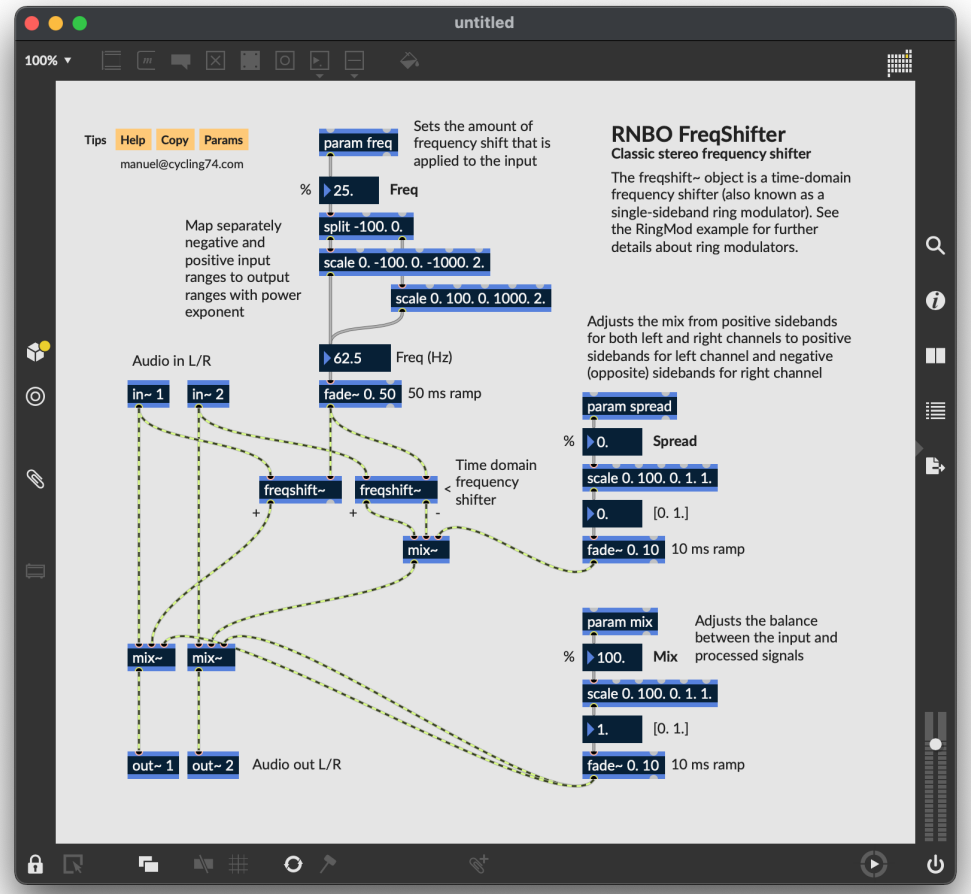


Fig. 6.52 **The RNBO FreqShifter patch.**

- Click the export icon on the right side of the Max patcher window. More information about RNBO export can be found here.
- You will now be able to select your H9000 as an export target. If it's not in the list, make sure that Emote can detect your H9000.
- Click the H9000 target to edit information such as the algorithm name, description, and algorithm ID. Enter an unoccupied ID between 10101 and 11999. If you select an ID that is already occupied by an algorithm, then that algorithm will be overwritten.

- Right-click the H9000 device and select “Export Target”. The algorithm will appear on your H9000 at the ID specified.
- Open Emote and load the algorithm, or select it from the front panel of your H9000.

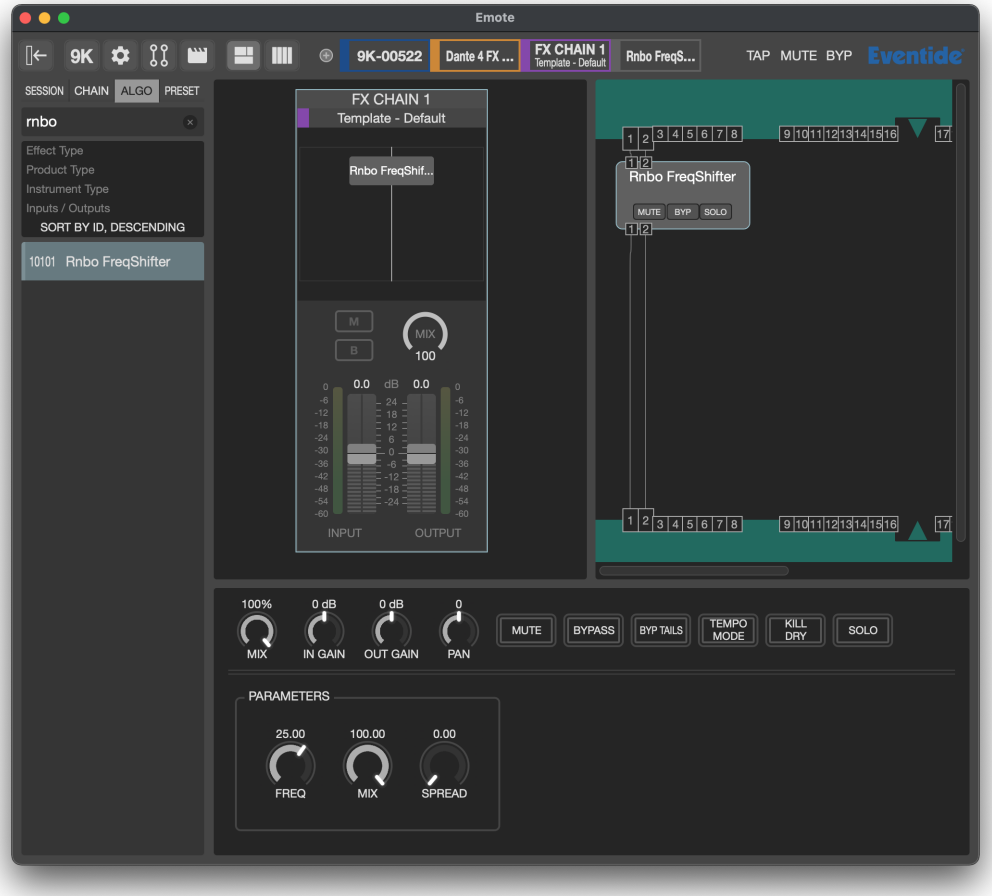


Fig. 6.53 **The RNBO freqshifter patch has been exported to the H9000 to algorithm ID slot 10101.**

- Note the parameters from the RNBO patch titled `param freq`, `param mix`, and `param spread` are now available as adjustable parameters within the H9000 algorithm.

Note

Your computer must be connected to both the H9000, and the internet to compile and export RNBO algorithms to the H9000.

6.3.3. Sharing RNBO Algorithms

If you would like to share your RNBO algorithm with other H9000 users, you may use the Dump Data feature to export a `.9ka` file that users can upload directly to their H9000 via Emote. To access the Dump Data feature, insert a FAT32-formatted USB drive into the H9000, then navigate to Emote's Devices View, click `UPDATES`, then click `DUMP DATA`. The `.9ka` file will be exported to the USB drive.

6.3.4. RNBO Limitations

- Retention of target settings per patcher (you'll have to reenter name/description/slot every time you export, sorry)
- buffer~/soundfile is not currently supported
- On-screen parameter display currently doesn't offer any customization.
- Some RNBO patches may need to be optimized to run properly on the H9000. Please post on the Cycling '74 forum if you are having trouble with a patch.

6.4. RNBO Tutorials

6.4.1. Max/RNBO Tips for the H9000 User

Max is a visual programming environment for DSP/multimedia projects. RNBO is a Max extension that allows users to build and export patches to external plugin formats.

Working in RNBO is very similar to Max, with some key differences. The first section of this guide will cover basic functionality shared by RNBO and Max. In the second section, we'll take a closer look at RNBO and how it works alongside Max.

This guide will cover the basics of Max signal flow and syntax, and give tips for the H9000 user who is familiar with VSIG looking to get started working in Max/RNBO. It is intended to help those with some DSP experience who are new to Max.

Tip

For more detailed guides, tutorials, and documentation for Max/RNBO, check out the resources over on cycling74's website:

- Forums
- Resources

To talk with other VSIG/H9000 users, check out the Eventide user forums at:

- Eventide Forums

Finally, Max's built in help and reference files are excellent resources for troubleshooting and finding examples for each object. To access them, right click an object and choose 'Open [object] help' or 'Open [object] reference.

6.4.1.1. Audio vs. Data

Like VSig, Max handles primarily handles two kinds of signal—audio and data signals. Audio signals are carried by striped green patch cables, while data signals are carried by gray patch cables. Objects that carry audio signals (MSP objects) end with the '~' symbol. See the examples below.

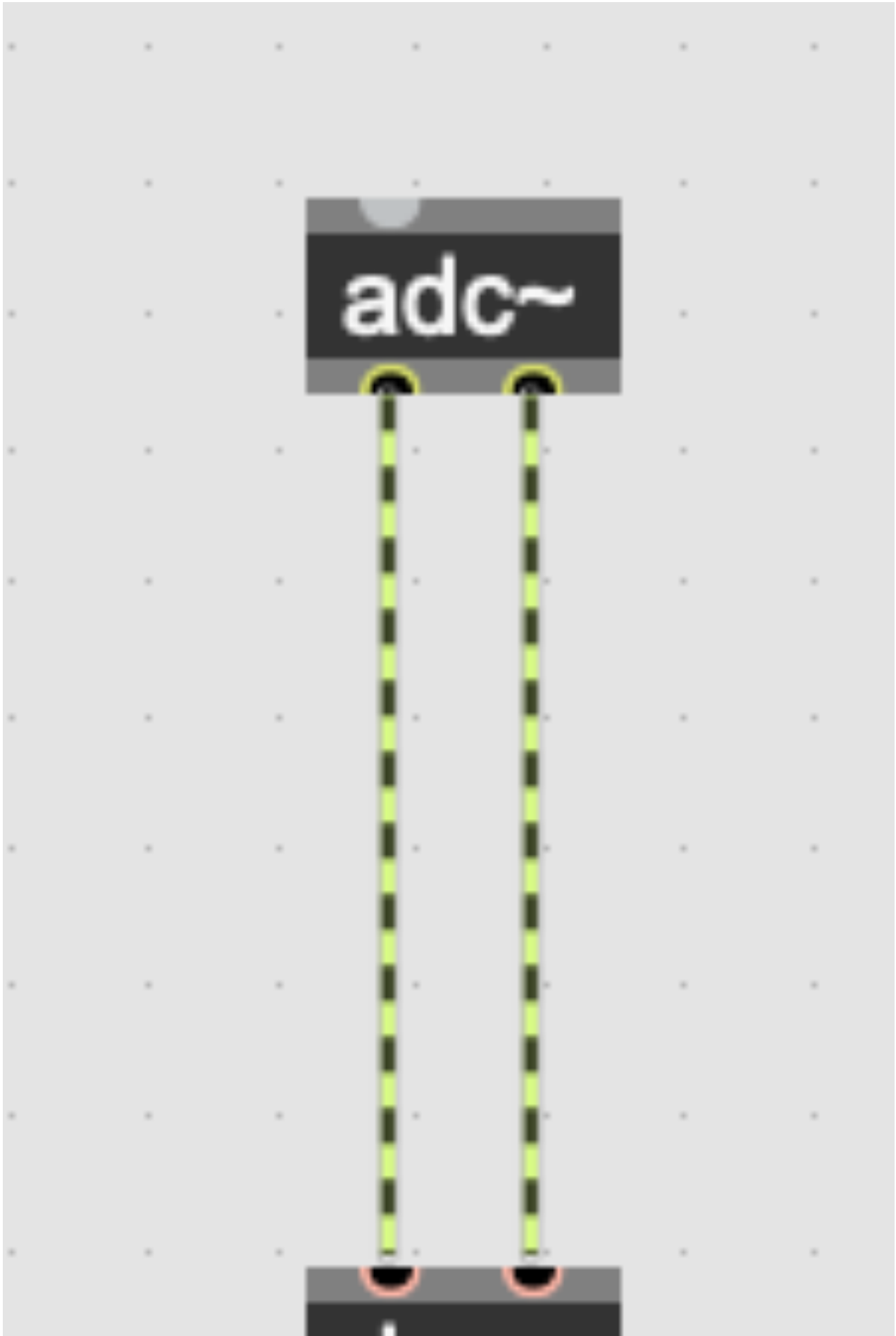

adc~

Fig. 6.54 **Audio signal cables**

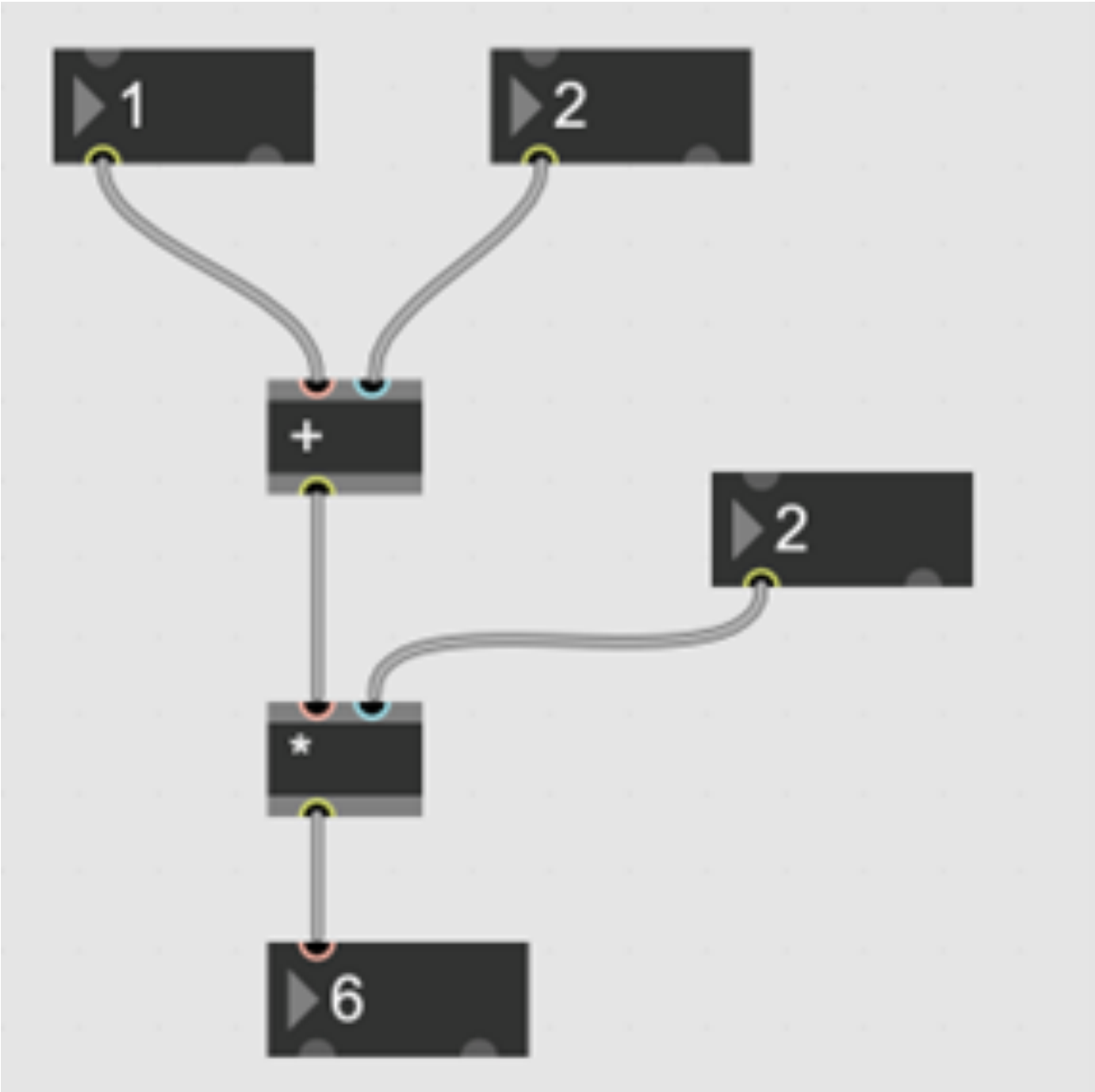


Fig. 6.55 **Data signal cables**

Keep in mind that many objects have multi-function inlets (often the inlet furthest left). Sometimes, the same inlet can take an audio and data signal at the same time.

Tip: To send/enable objects to accept floating points, a decimal point must be included at the end of your argument. Otherwise, Max will interpret it as an integer.

6.4.1.2. Bang

The ‘bang’ message is one of the most important features in Max. When an object receives a bang, its output is triggered. Bangs can be sent in a number of ways, the simplest being the bang button, a gui object that bangs when clicked. Many objects also output a bang when once their operation is complete. This is especially useful for creating networks of ordered events and processes in your patch.

Fig. 6.56 **Button that outputs a bang when pressed**

A bang produces a single instantaneous trigger. For switching and holding (for example, turning on the **[count]** object), objects take a binary value.

6.4.1.3. Audio Routing

To select your audio device in Max, go to Options>Audio Status. From this menu, you can change your input and output devices, adjust the sample rate and i/o vector, and customize your channel routing.

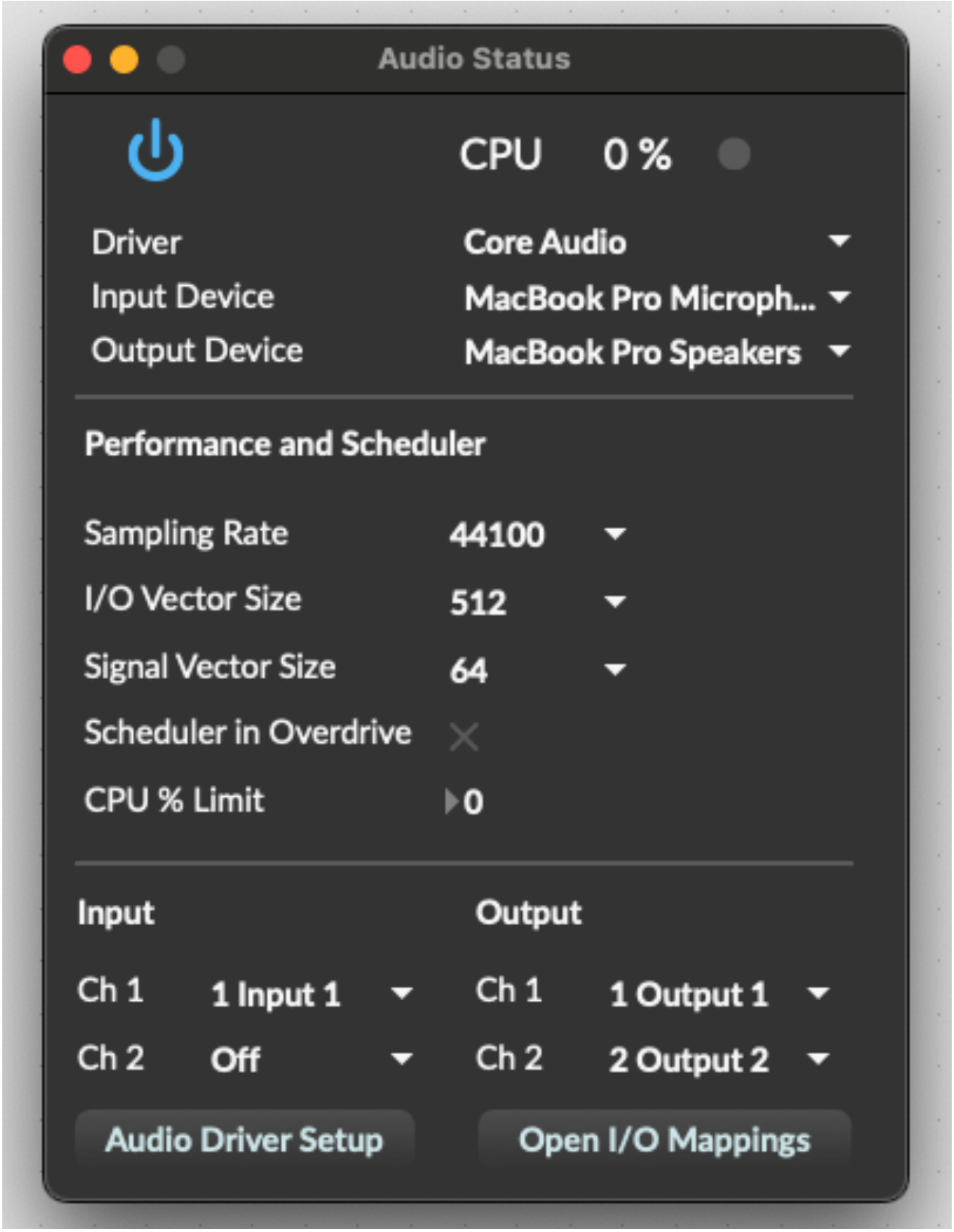

Audio Status
CPU 0 %
Driver
Core Audio
Input Device
MacBook Pro Microph...
Output Device
MacBook Pro Speakers
Performance and Scheduler
Sampling Rate
44100
I/O Vector Size
512
Signal Vector Size
64
Scheduler in Overdrive
CPU % Limit
0
Input
Output
Ch 1
1 Input 1
Ch 2
Off
Ch 1
1 Output 1
Ch 2
2 Output 2
Audio Driver Setup
Open I/O Mappings

Fig. 6.57 **Audio Status window**

> **Note**
>
> Max does not automatically follow your current system audio routing. If your system is connected to an external audio interface and you want to connect Max to that interface, you must do it manually each time you launch Max. Typically, Max will connect to your internal computer mic and speakers by default. If you forget to change the device, this can lead to some nasty feedback.

The *[adc~]* and *[ezadc~]* objects bring audio into Max. The *[adc~]* object must be given arguments for each channel. By default, *[adc~]* is set to channels 1 and 2. If you wanted it to take channels 1-4, you would enter the channel numbers as sequential arguments:

Fig. 6.58 ***[adc~]* objects**

[ezadc~] is a gui object that inputs only from channels 1-2.

Fig. 6.59 ***[ezadc~]* object**

The same applies to the audio output objects, *[dac~]* and *[ezdac~]*. All four of these objects can be used to turn audio on/off, either with a binary input or by clicking on the gui objects. The button in the bottom right of the Max window is blue when audio is on, and gray when it is off.

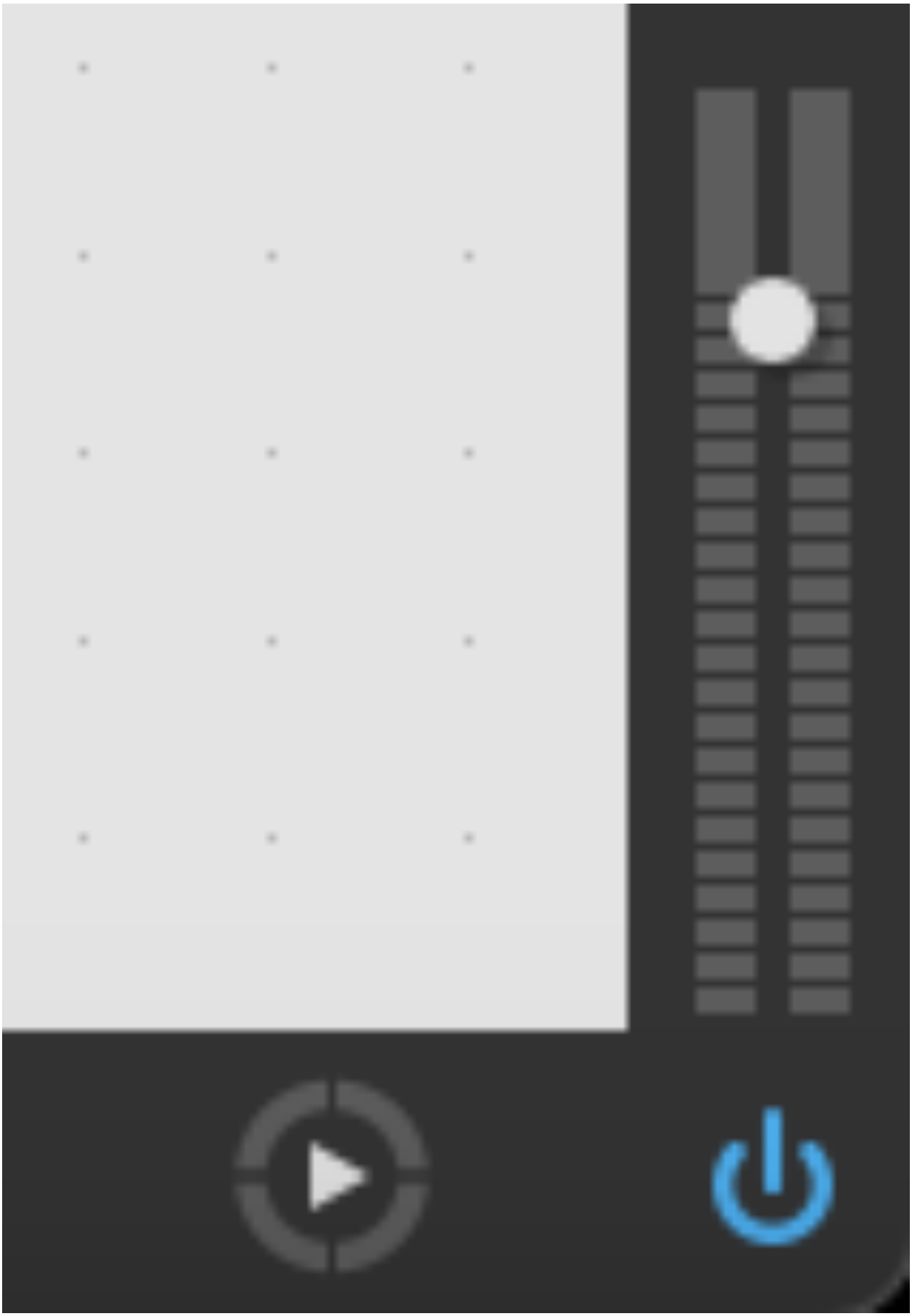


Fig. 6.60 **Audio on, indicated by blue power button**

6.4.1.4. Patch Cables

Objects are connected in Max using virtual patch cables. To do this, click and hold on an object outlet and drag it over to an inlet. To make connections without patch cables, you can use *[send]* and *[receive]* objects. These are helpful for sending a message to many places at once, as well as communicating between subpatchers.

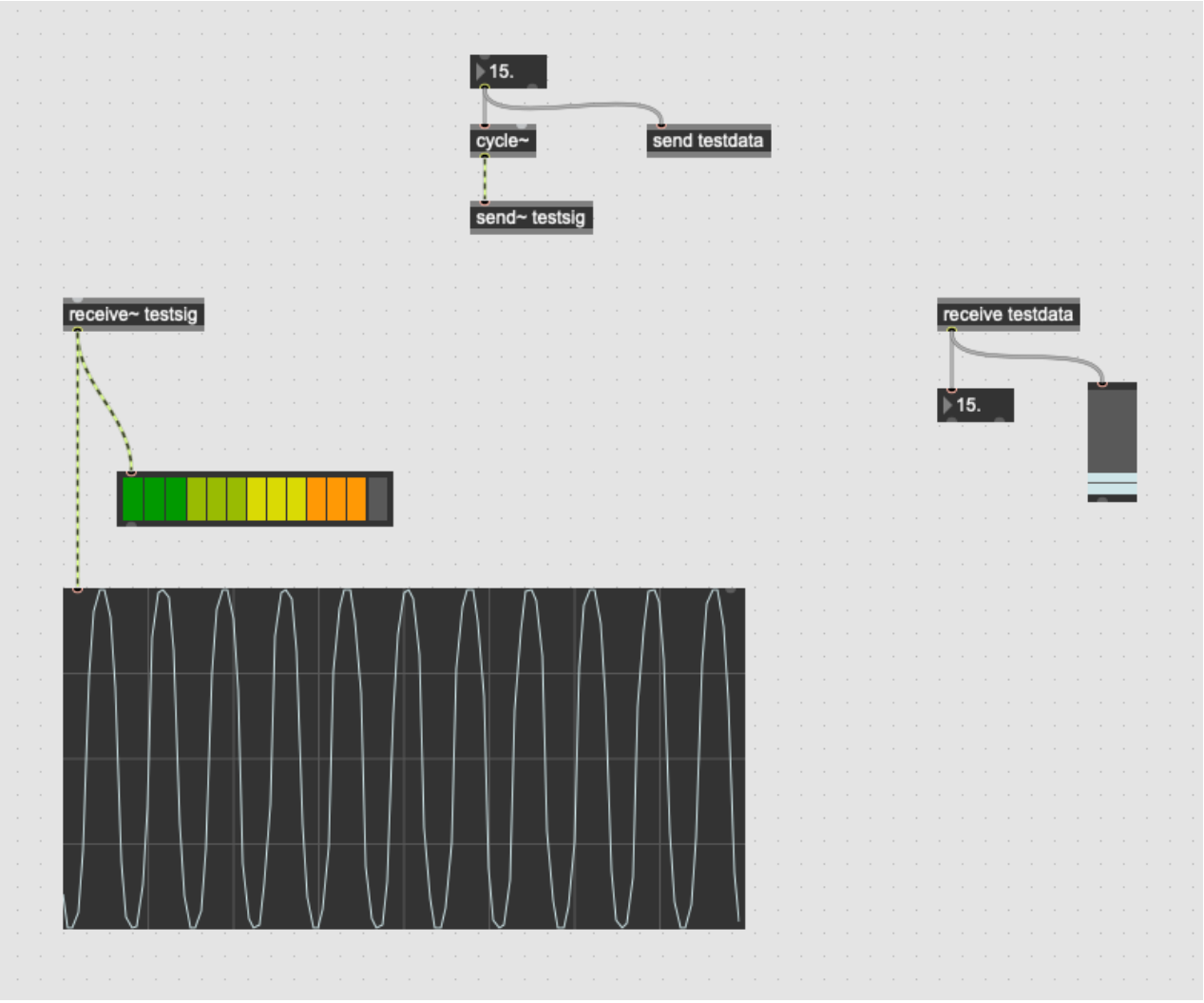


Fig. 6.61 **Basic Max patch demonstrating *[send]* and *[receive]***

For convenience, send and receive objects can be abbreviated as *[s~]*, *[r~]*, *[s]*, and *[r]*.

6.4.1.5. The Console and Inspector Windows

The console and inspector windows are important tools for debugging your patch and adjusting advanced object parameters/UI settings. They can be opened via the right menu pane in Max and RNBO. Inspector can also be opened with the **cmd+i** (Mac) or **ctrl+i** (Windows) keyboard shortcuts.

Fig. 6.62 **Console**

Fig. 6.63 **Inspector**

The Max console displays various system messages, including errors, package information, and values printed by the *[print]* object. The console is useful for testing and debugging your patches.

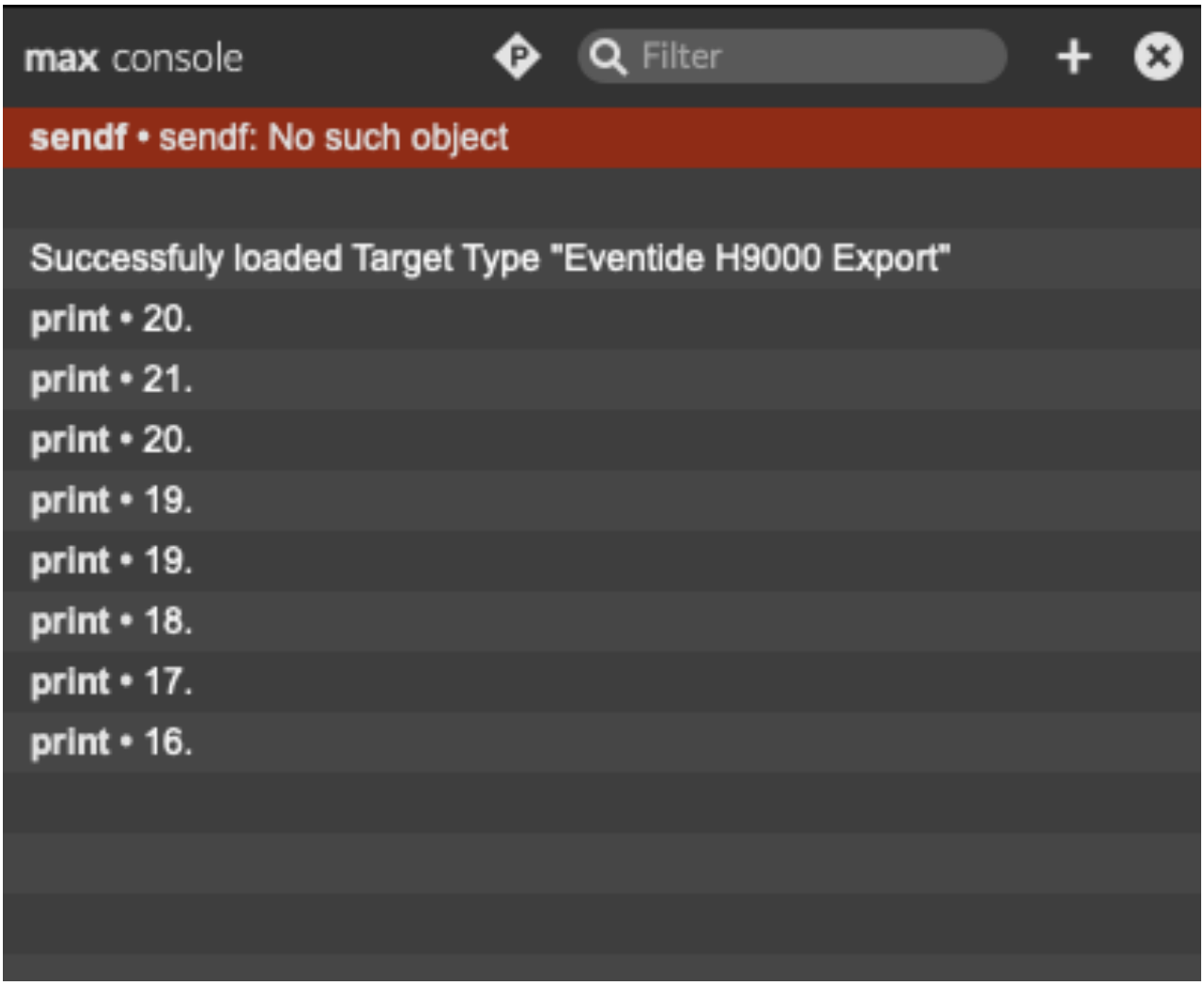


Fig. 6.64 **Console window**

The inspector window lets you edit various functional and UI settings for Max objects. Select an object in Max, and its settings will appear in the inspector window. In the example below, you can change settings for the number object, including its min and max values, font, size, color, and more.

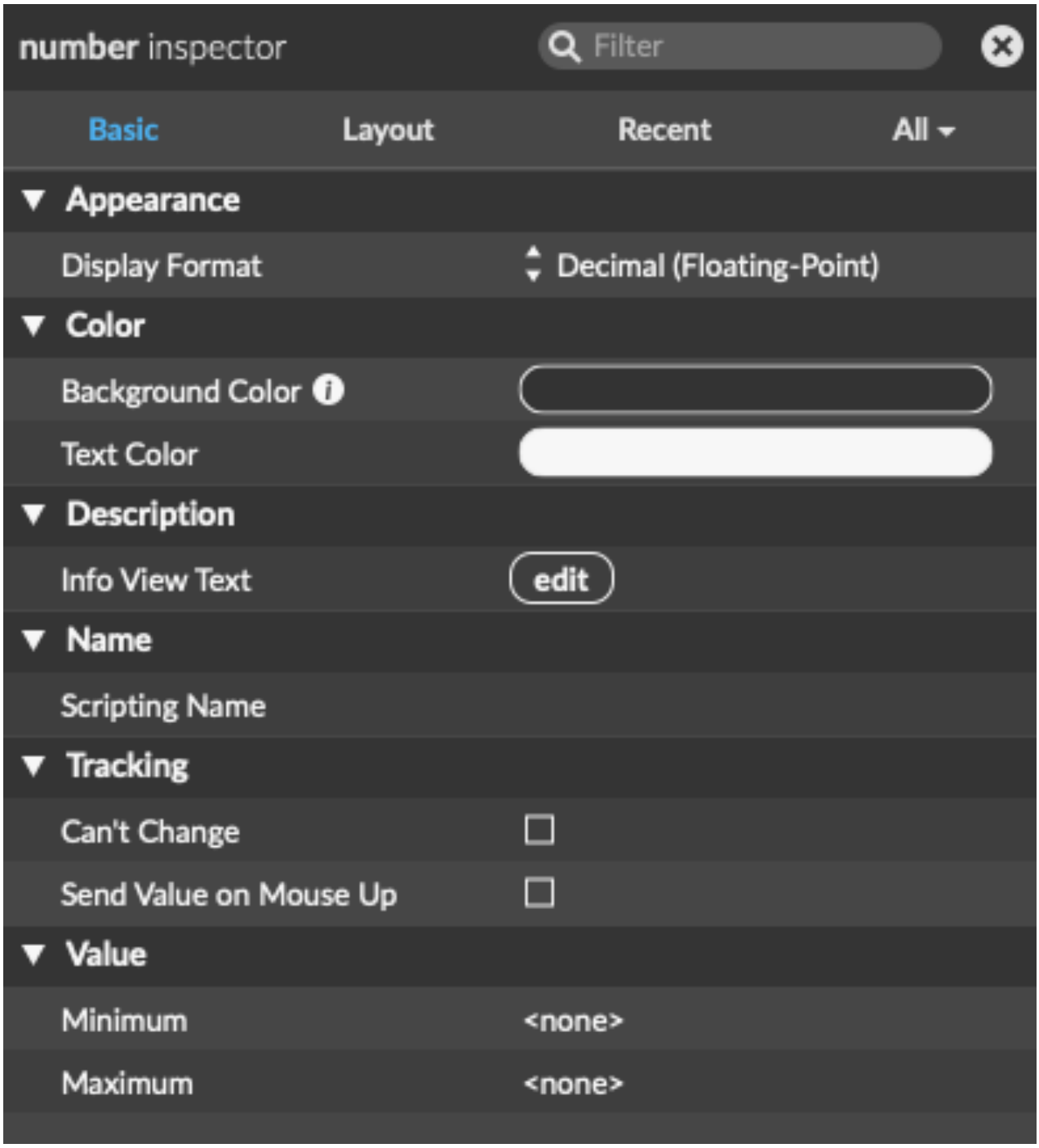


Fig. 6.65 **Inspector window**

6.4.1.6. RNBO Signal Flow

To create a RNBO subpatcher, enter the *[rnbo~]* object in Max. This will generate a new RNBO patching window. RNBO objects are distinguished visually from Max objects by their blue border.

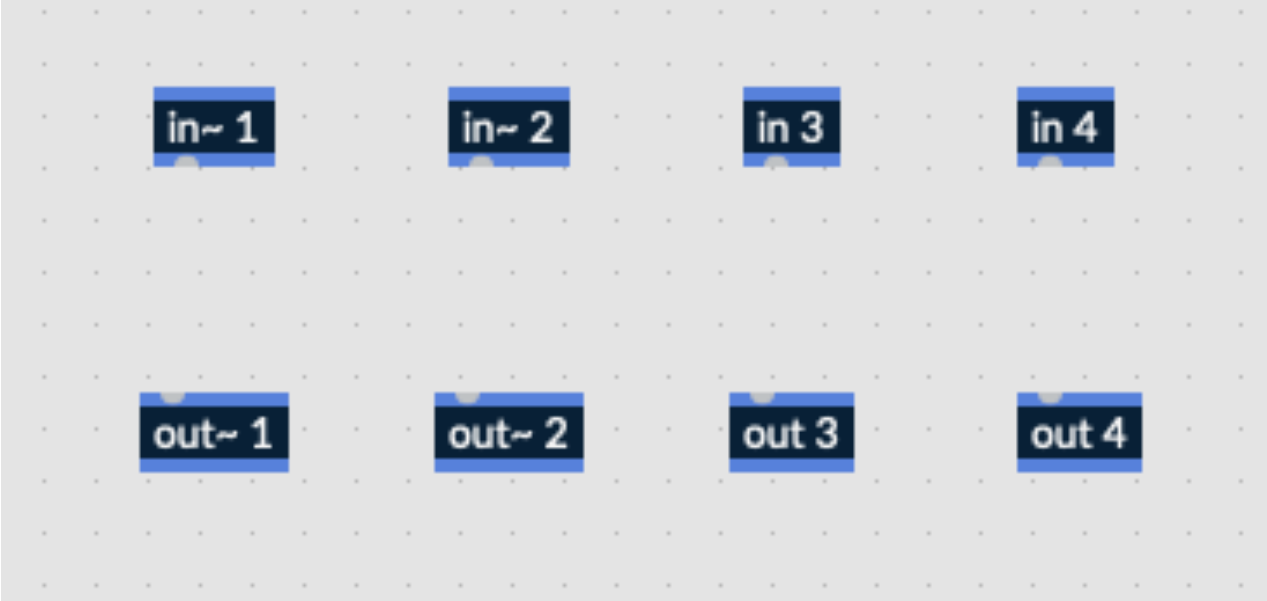


Fig. 6.66 **RNBO objects with blue borders**

RNBO does not have built in adc/dac objects. Audio must be routed from the container Max patch into RNBO. To do this, you'll need to create *[in~]* and *[out~]* objects in RNBO. A positive integer argument must be defined, and arguments cannot be repeated, even between audio and data ins. For example, you can't have *[in~ 1]* and *[in 1]* in the same RNBO patch. These ins and outs will appear as inlets and outlets on the *[rnbo~]* object in Max.

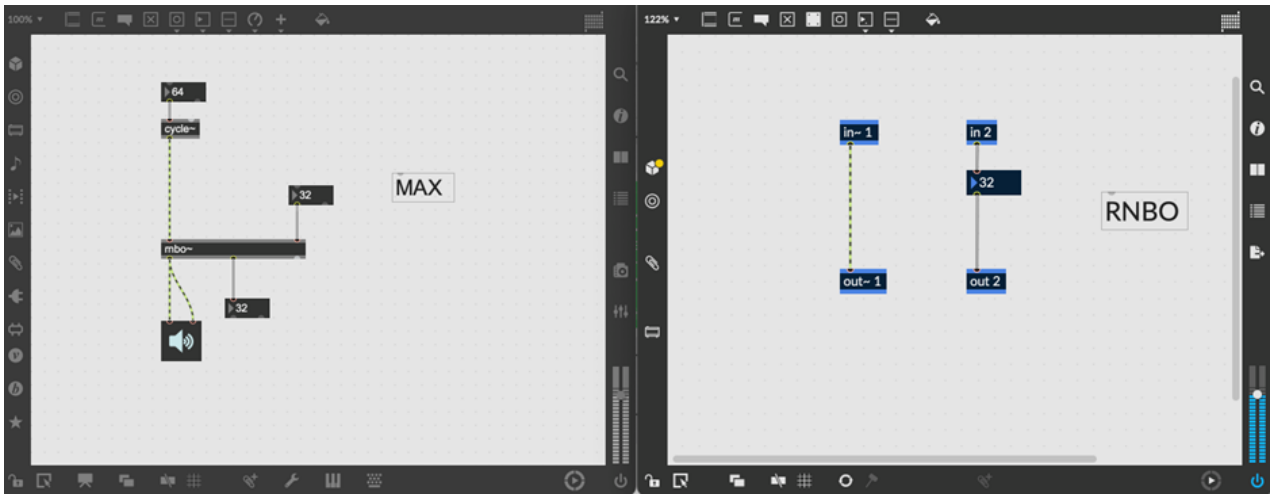


Fig. 6.67 **Max objects with gray borders shown on left. RNBO objects with blue borders shown on right.**

6.4.1.7. Using Parameters

One of the most important tools in RNBO is the *[param]* object. This object allows you to define parameters in your RNBO patch and control them externally. Parameters are converted to controls in Emote when exporting RNBO patches to the H9000.

> **Note**
>
> Without parameters, you will not be able to adjust any controls on your patch once it is exported to the H9000.

To create a parameter, create a *[param]* object in RNBO with a name argument:

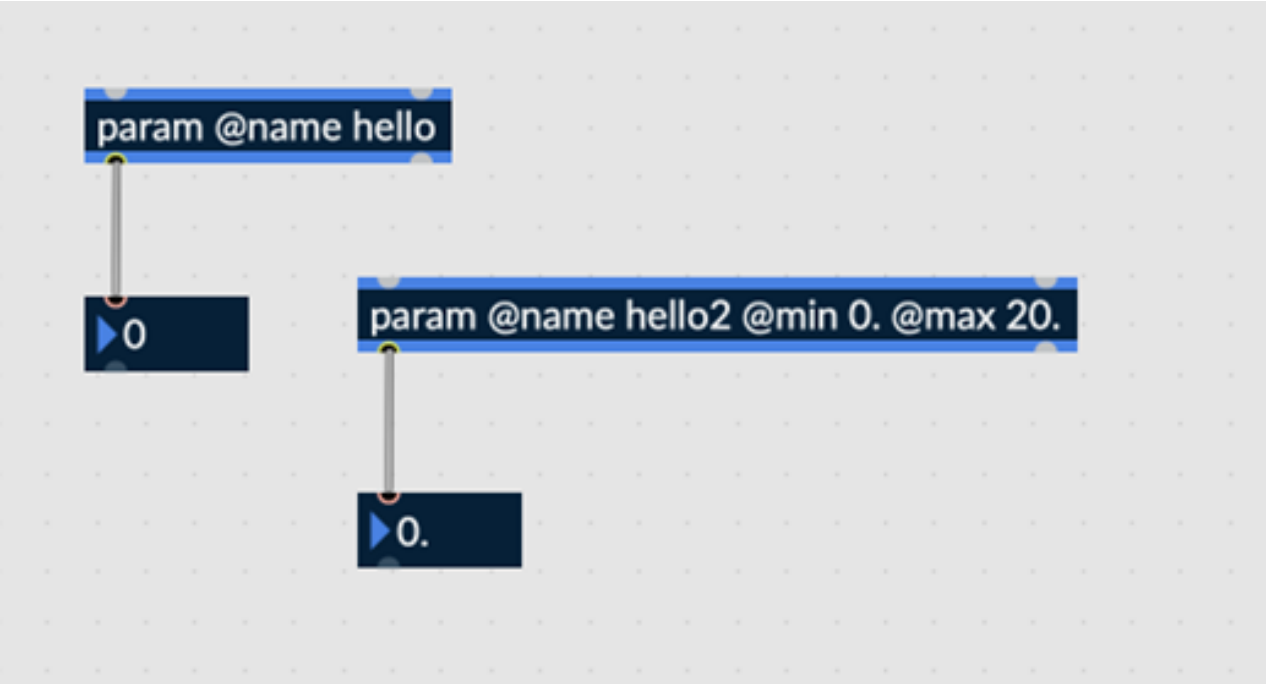


Fig. 6.68 **Two *[param]* objects, named "hello" and "hello2"**

You can also define arguments for the minimum and maximum values the parameter will accept. In the example above, the 'hello2' parameter is set to accept float values between 0 and 20.

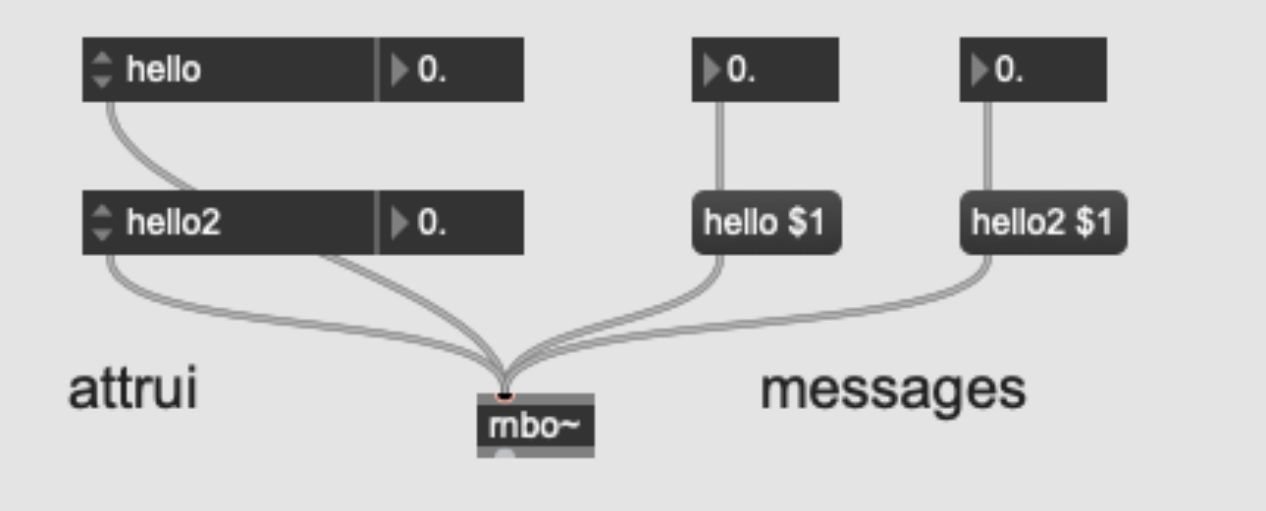


Fig. 6.69 **Controlling parameters in Max**

The simplest way to control a RNBO parameter in Max is with the *[attrui]* object. When connected to the *[rnbo~]* object, it will show all available parameters in its dropdown menu. You can also send RNBO a message with the parameter name and value. In the example above, the float boxes send a value to the $1 variable object in each message box and trigger the message output. You can see that the values have been received in the RNBO patch.

For more information on parameters, refer to the *[param]* help patch in RNBO.

6.4.2. RNBO Tutorial #1 - Simple Gain

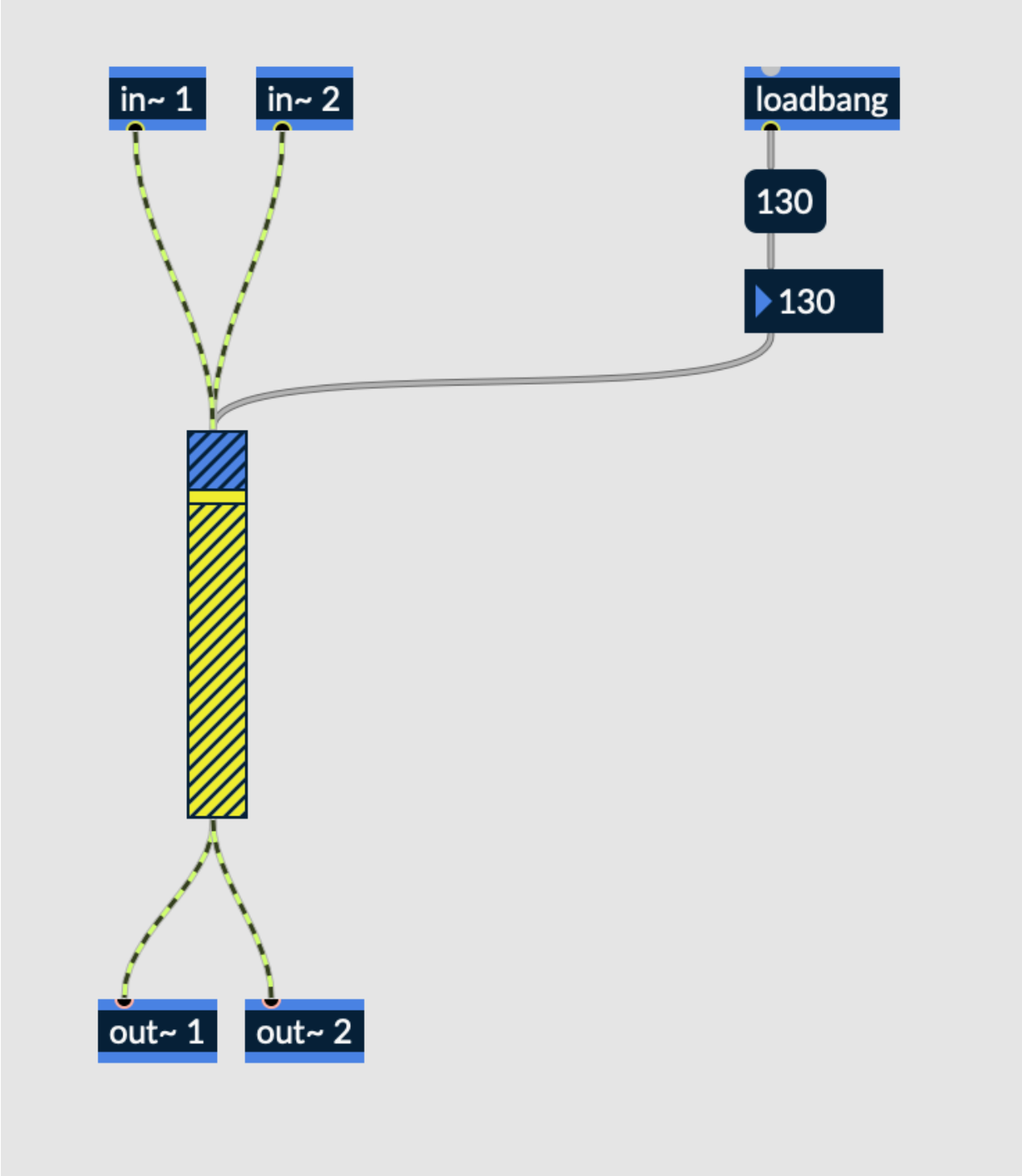


Fig. 6.70 **Simple gain patch in RNBO**

The above patch is designed to show how audio signal is routed in RNBO using a simple gain object. This patch includes no parameters that the user has any direct control over upon exportation to Emote. Despite the patch's limited functionality, hopefully it will serve as an effective introduction to routing signal in the RNBO environment.

Download the maxpat file here: Simple Gain.

6.4.2.1. Inputs and Outputs

When instantiating the *rnbo~* object in Max, no signal inlets or outlets are created by default. One must create the desired amount of *in~* and *out~* objects within the RNBO subpatcher, after which the corresponding number of inlets/outlets will appear on the *rnbo~* object.

Fig. 6.71 **RNBO~ object, with two signal inlets and outlets**

In this case, two *in~* and two *out~* objects exist within the RNBO subpatcher, so two signal inlets and two signal outlets appear on the *rnbo~* object in the main patcher.

Upon exporting to Emote, the algorithm will also have two signal inlets and two signal outlets, viewable within the FX chain window.

Fig. 6.72 **After exporting, the algorithm can be added to the Emote FX chain with the same amount of inputs and outputs as in RNBO**

6.4.2.2. Gain~ Object

Max/MSPs *gain~* object is shown to the user as a slider or fader, and provides a set amount of gain to an incoming signal. *Gain~* can access values from 0-157. **These values are not directly equivalent to gain in decibels.** In this case, the gain is set to a constant value of 130, which is the equivalent of roughly 6 dB of gain. A value of 128 represents unity gain.

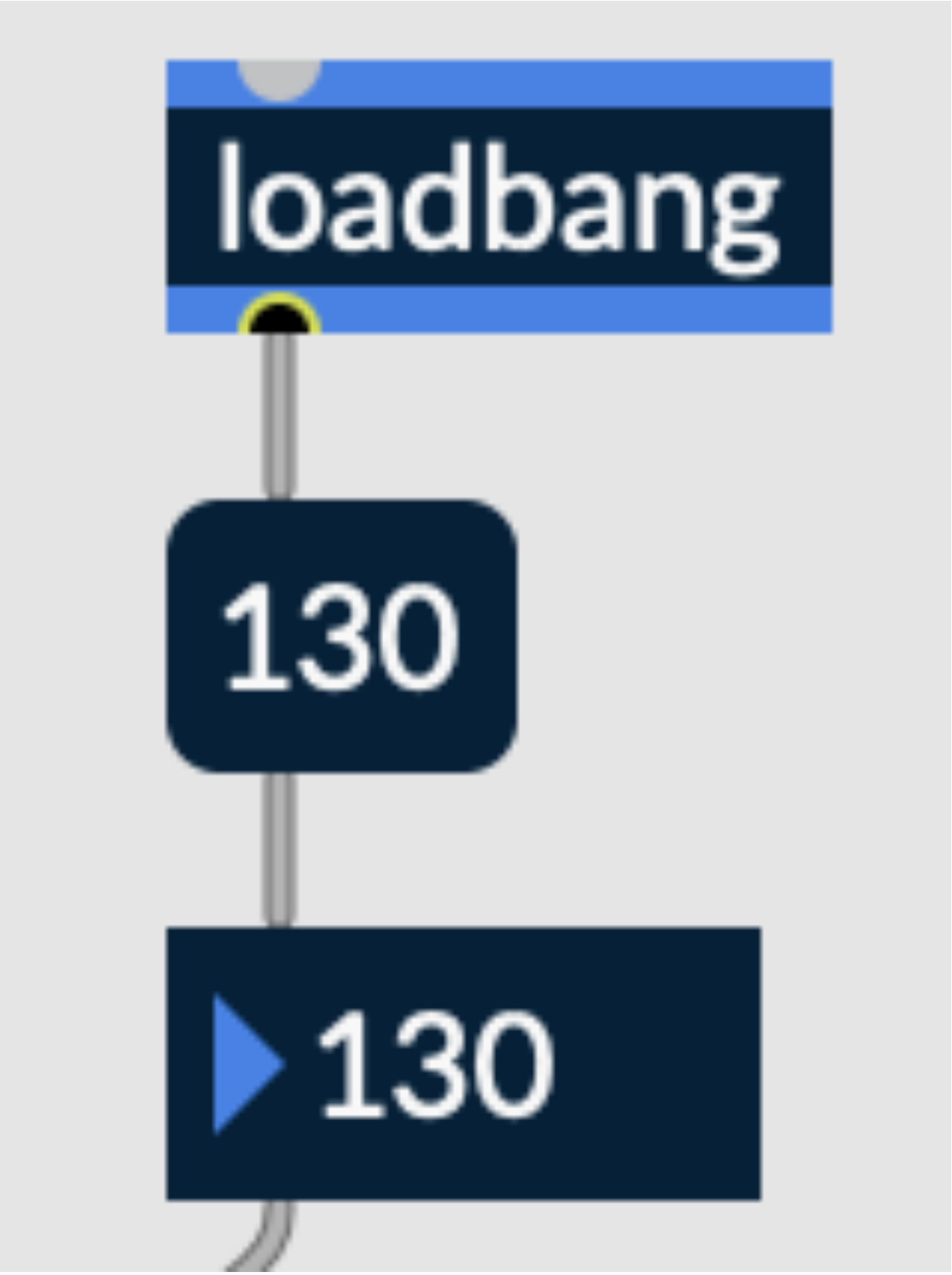

loadbang
130
130

Fig. 6.73 **The above objects are used to set the gain to 130 upon loading the patch**

Loadbang is used to ensure that every time the patcher is loaded (or uploaded to the H9000), gain~ is set to a value of 130.

6.4.2.3. Testing the Patch

One advantage to using RNBO for algorithm development is the ability to test effects directly within Max without having to reupload to a hardware device every time a change is made. The main Max patch for this simple gain algorithm is shown below:

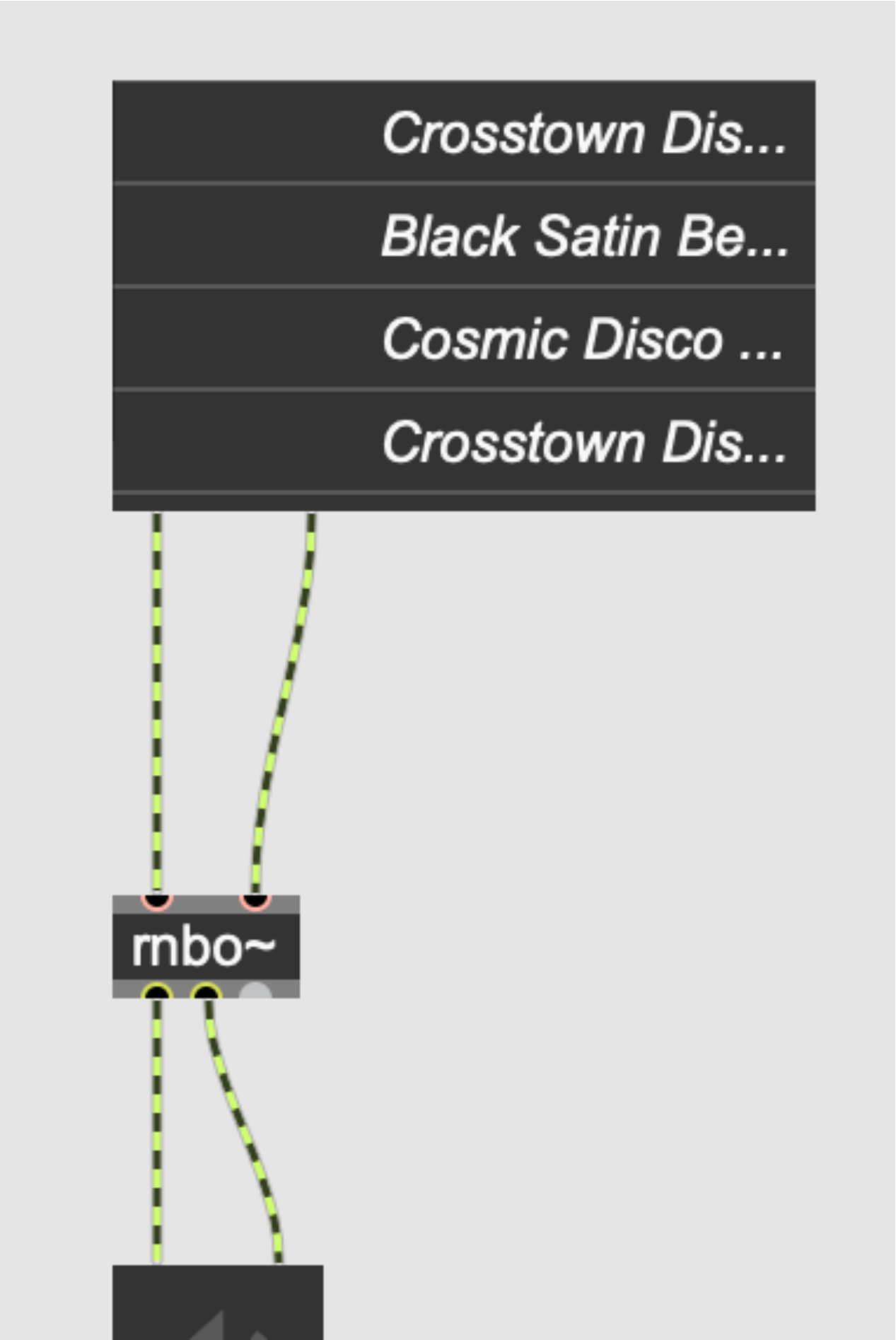

Crosstown Dis...
Black Satin Be...
Cosmic Disco ...
Crosstown Dis...
rnbo~

Fig. 6.74 ***Playlist~* object used to test patches within Max**

Both signal outlets from *rnbo~* are routed to *ezdac~*, allowing audio to be monitored through your computer speakers or any audio device. The *rnbo~* signal inlets receive audio signals from a *playlist~* object containing four short audio files. *Playlist~* makes it very quick and easy to test a RNBO patch on a variety of audio files of your choosing, although *ezadc~* can also be used if you'd like to play an instrument through your effect.

Upon exporting to Emote, the algorithm will appear in the algorithm location designated by the user when exporting from Max and must be dragged to the FX chain to be used. The algorithm has no user controls, but provides a constant amount of gain to an incoming signal (roughly 6 dB).

6.4.3. RNBO Tutorial #2 – Simple Gain with UI

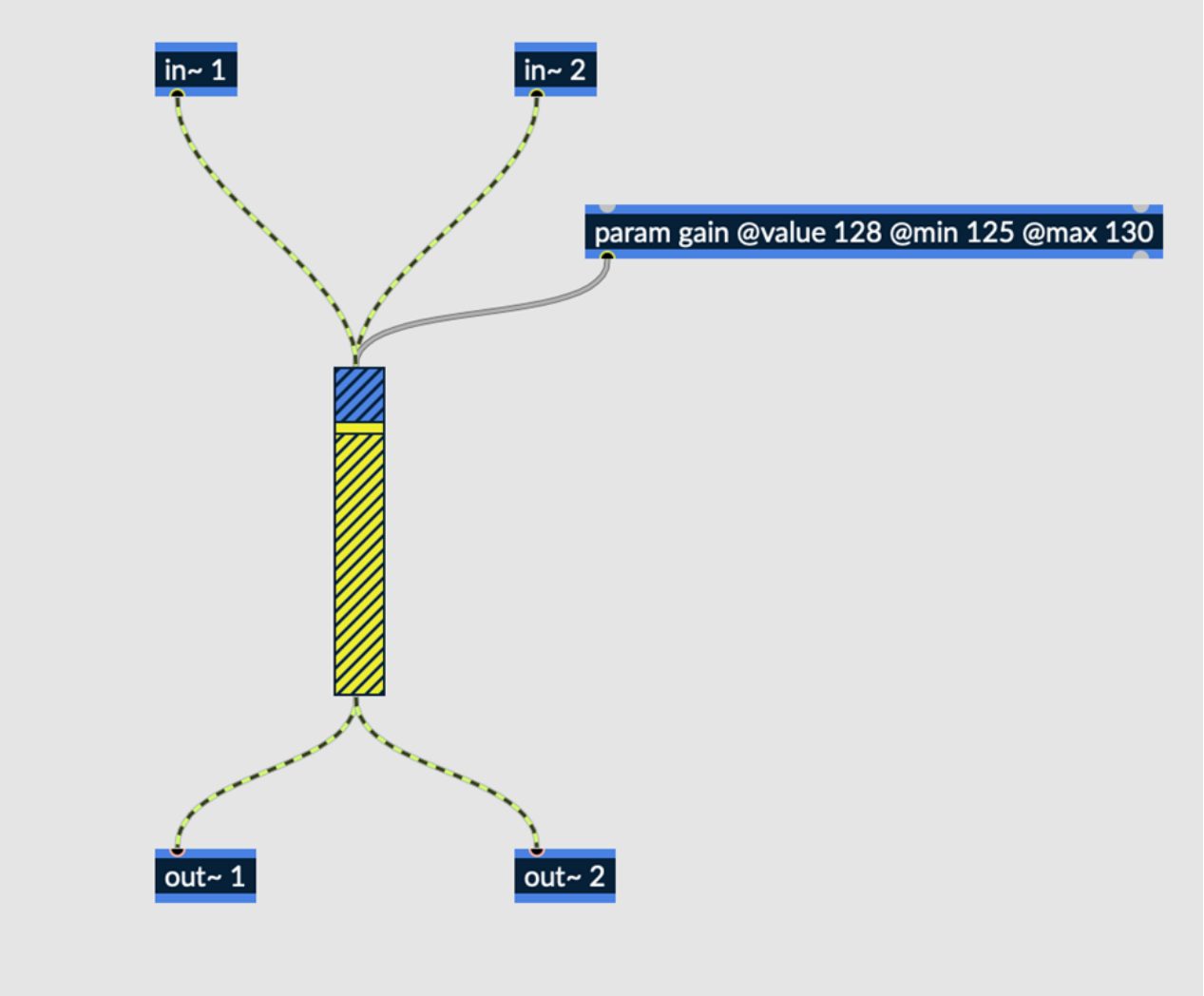


Fig. 6.75 **Simple gain patch with UI in Max**

The above patch is an expansion upon the patch from Beginner Tutorial #1, providing the user with manual control over gain. This tutorial will explore not only how to control gain within the main Max patcher, but also how to do so after exporting to Emote.

Download the maxpat file here: Simple Gain with UI.

6.4.3.1. Param Object

The *param* object is the primary means by which the end user is given control over certain parameters of a patch. Without param objects, the only way to alter the parameters of an effect is directly within the RNBO subpatcher.

Param objects are instantiated within the RNBO subpatcher, and effectively act as variables which can have values sent to them from outside the RNBO subpatcher.

Fig. 6.76 ***Param* object used to control gain**

Param objects require a name and initial value (@value) as arguments, although many other attributes are available. In the example above, the @min and @max attributes are also used, setting the minimum and maximum possible values the object can send.

6.4.3.2. Controlling Parameters within the Max Patcher

Once a *param* object is instantiated within RNBO, there are a couple ways to control the parameter in Max.

Attributes Menu

- Right click any signal inlet on the *rnbo~* object to view its attributes. All instantiated parameters should be included in this list as attributes.
- Click on any available parameter to open its corresponding *attrui* object.
- Control over a given parameter is accessible directly within its *attrui* object. The way in which *attrui* displays data can be changed in the Inspector, in the “Display Mode” field.

Message Box

- Values can also be sent to a given parameter in RNBO using a *message* box with the same name as the parameter. An example of this is shown below:

- A *message* with the argument "gain $1" is connected to the first signal inlet of the *rnbo~* object, allowing values to be sent to the gain parameter.
- "$1" effectively acts as a placeholder, containing whatever value is most recently sent to the message by the number box above.
- The value of the number box is controlled by a slider. Setting up parameter control in this way takes slightly more effort than doing so via the attributes menu but allows for greater customization over exactly how the user controls parameters.

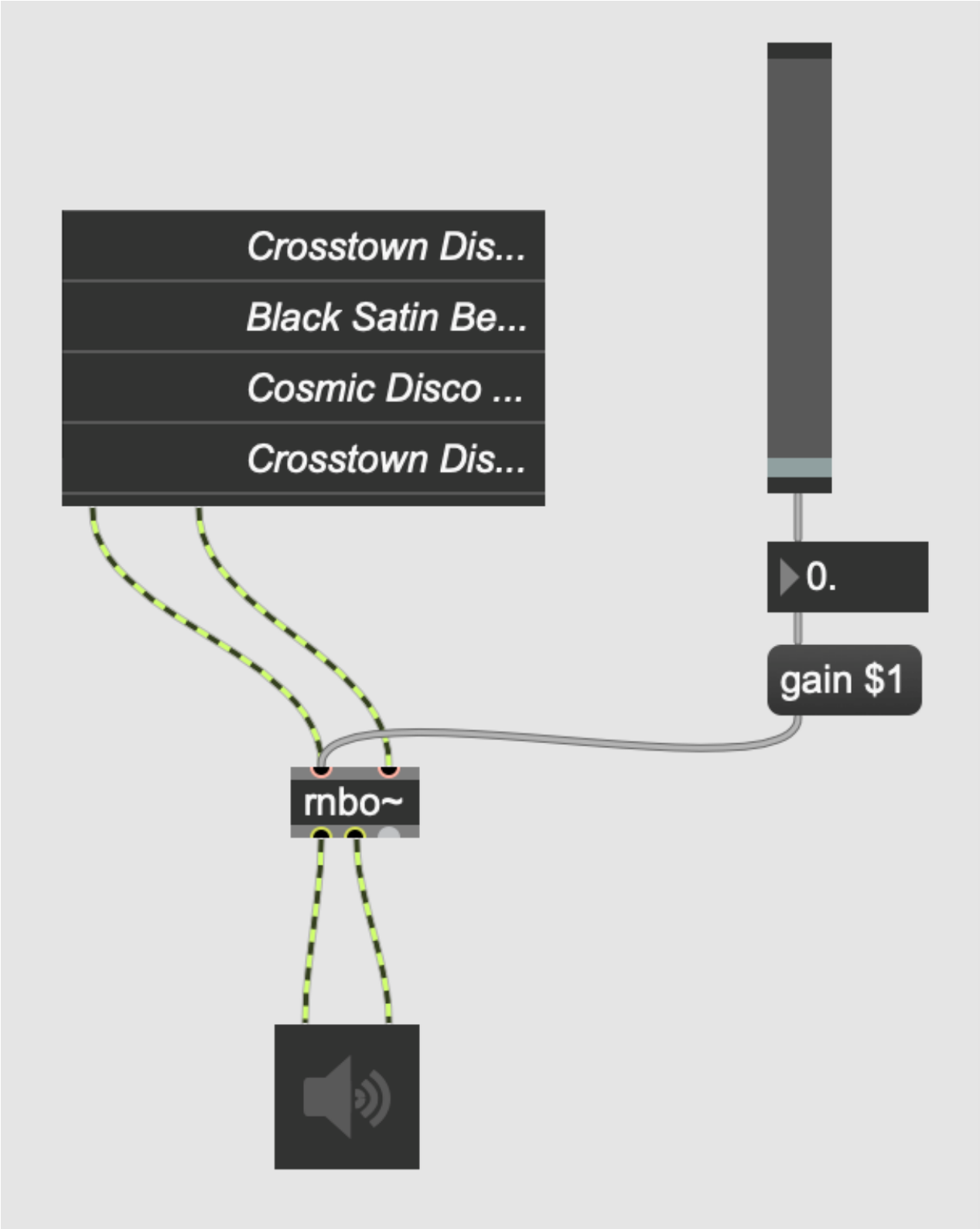


Fig. 6.77 **Example of how to control parameters within Max using messages**

Fig. 6.78 **UI shown in Emote, one knob to control gain**

Upon exporting to Emote, a single knob labeled “gain” will be presented to the user. A knob is automatically generated for every param object instantiated within RNBO and can only access values between the min and max values specified within the object. These values are displayed to the user above

the knob. Recall that the values accepted by RNBO's *gain~* object are NOT equal to gain in dB. A value of 128 is roughly equal to unity gain.

6.4.4. RNBO Tutorial #3 – Simple Gain with Filters

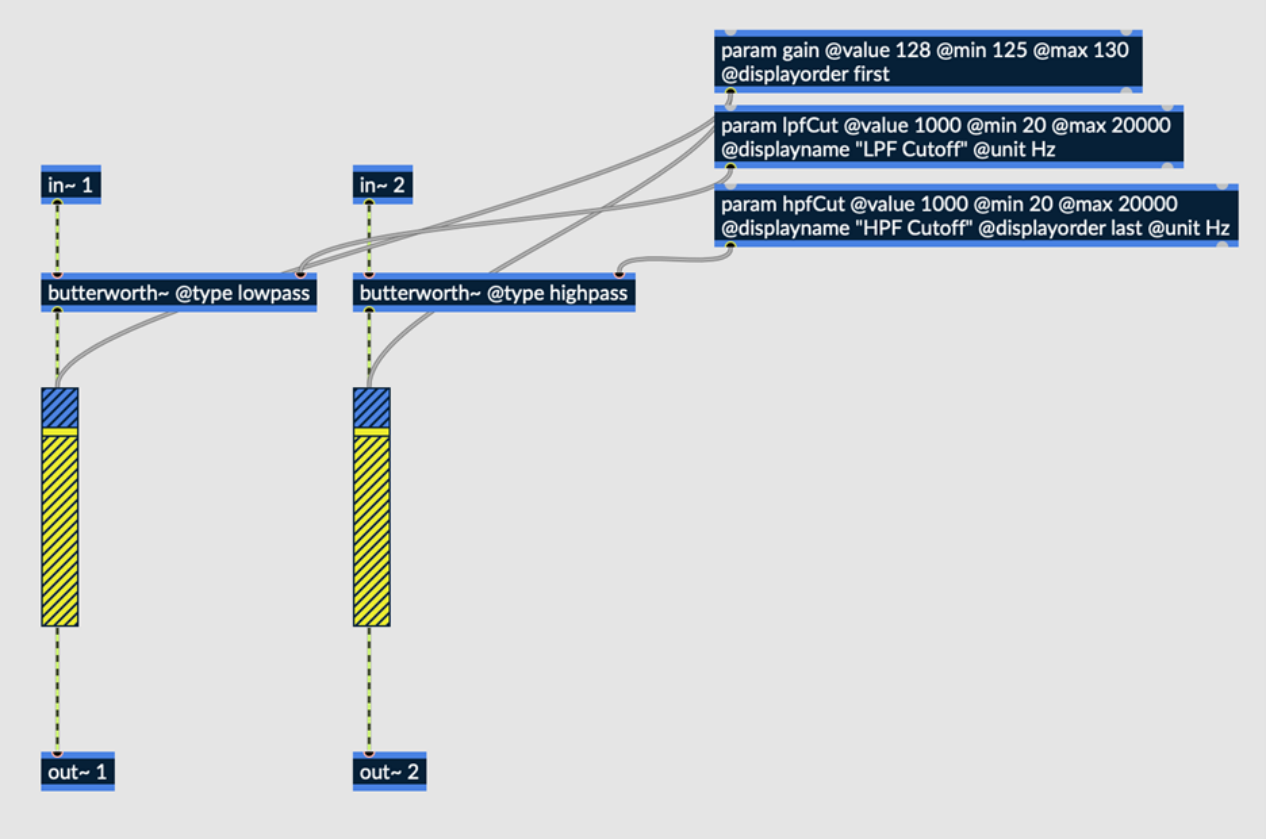


Fig. 6.79 **Simple gain, with low and high pass filters applied to the left and right channels respectively**

The above patch continues to expand upon Beginner Tutorials #1 and #2, this time demonstrating how each audio input may be processed completely independently of the other. The *butterworth~* filter object in this case is used to process input 1 with a low pass filter, while input 2 is processed with a high pass filter.

Download the maxpat file here: Simple Gain with filters.

6.4.4.1. Butterworth~ Objects

Butterworth~ is a filter object which can act as either a low or high pass filter depending on its @type attribute. Butterworth refers to a type of analog filter with maximally flat frequency response in the pass band.

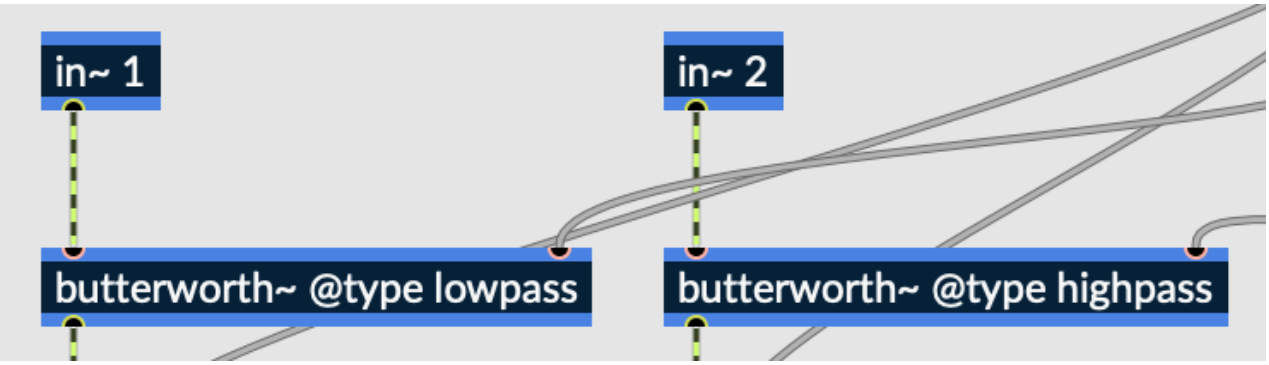


Fig. 6.80 **Inputs 1 and 2 both routed into *butterworth* objects**

In this patch, input 1 is processed by a *butterworth~* object with a "lowpass" @type attribute, meaning it acts as a lowpass filter. Similarly, input 2 is processed by the same object with a "highpass" @type attribute, meaning it acts as a highpass filter.

While maybe not the most practical effect, the goal of this patch is to demonstrate how each audio input may be processed completely separately from and independently of the other inputs.

6.4.4.2. Gain~ Objects

Unlike the first two beginner tutorials, this patch utilizes two identical *gain~* objects, one for each input. Now that each input is being processed differently, they must both have their own *gain~* object to avoid summing both inputs to a single output signal. Notice, however, that the gain *param* object simultaneously controls both *gain~* objects, providing the end user with a single gain control for both channels.

6.4.4.3. Param Objects

Two *param* objects have been added to the patch, both controlling the cutoff frequency of each filter object. As you may have noticed, new arguments are being used on these *param* objects. Let's look a little deeper:

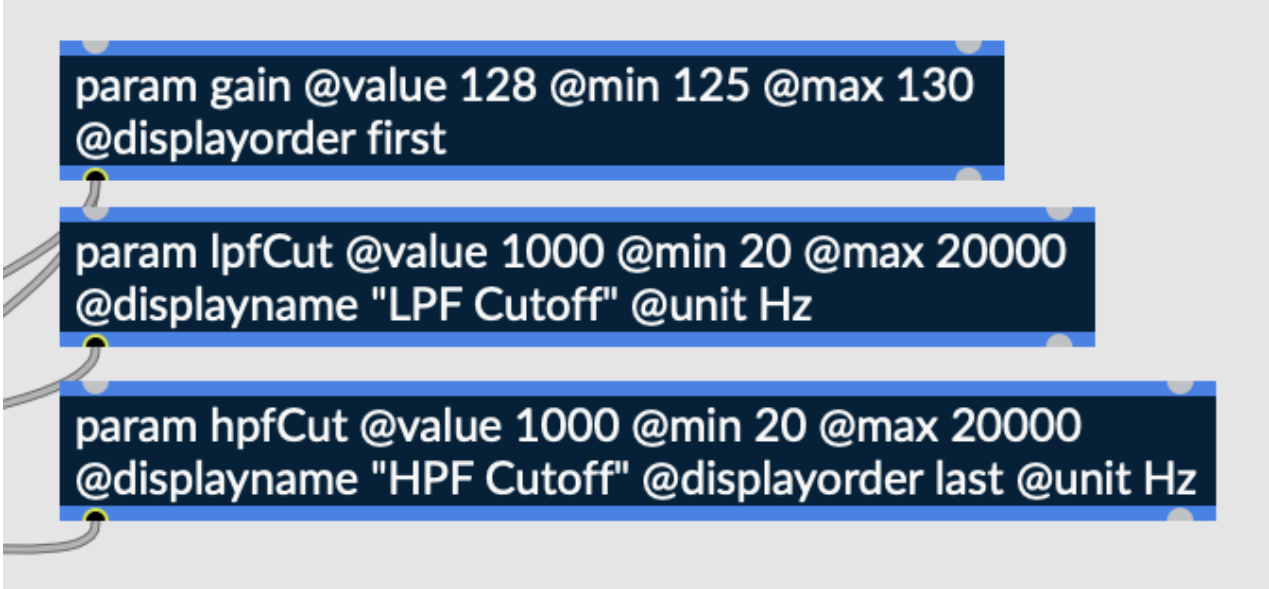


Fig. 6.81 ***Param* objects, controlling gain, LPF cutoff frequency, and HPF cutof frequency**

As previously discussed, @value, @min, and @max designate the parameter's default, minimum, and maximum value respectively.

Displayname

@displayname is used to designate how the knob will be labeled in Emote, or in any external environment which the patch has been uploaded to.

Displayorder

@displayorder is used to designate the order in which parameters will appear in an external environment. Order can be specified numerically, or symbolically using the keywords 'first' and 'last.'

Unit

@unit is used to describe the unit of a parameter in an external environment and will be displayed next to its numerical value.

With all these attributes being used, the UI within Emote looks like this:

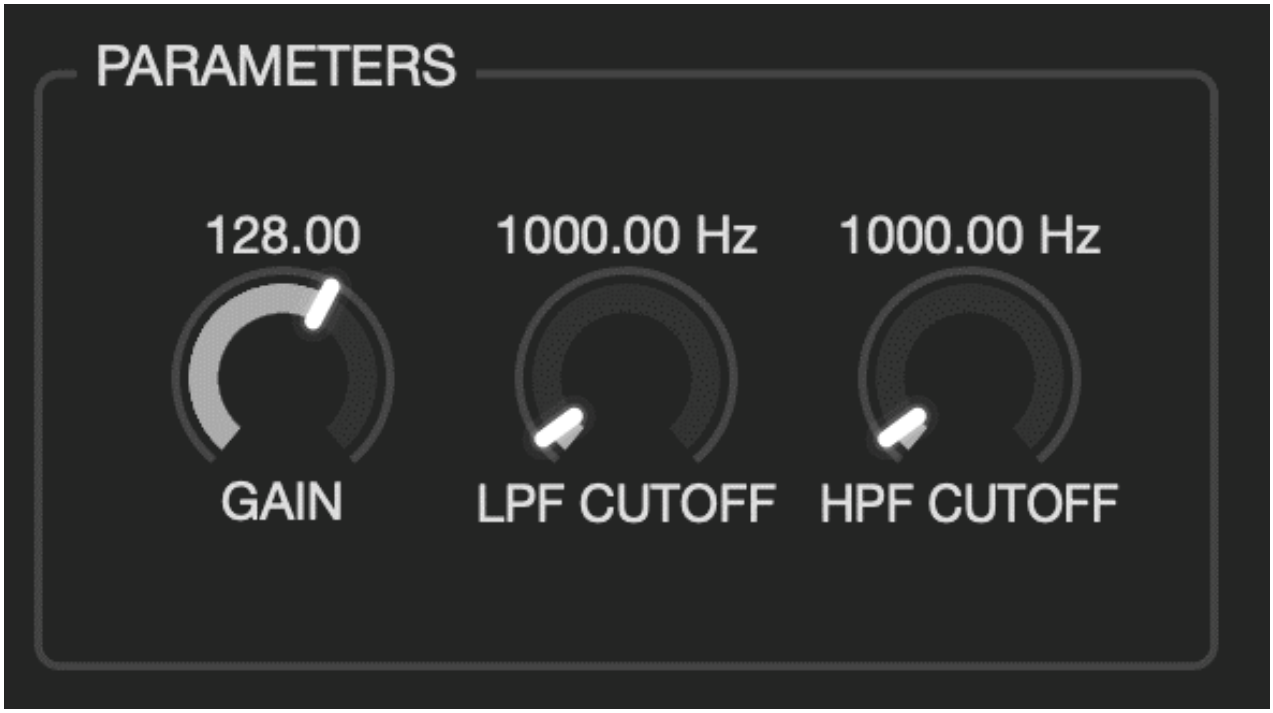


Fig. 6.82 **UI displayed in Emote, containing 3 knobs**

Notice that the filter cutoff frequency knobs are labeled using their @displayname attribute, as opposed to the name of their *param* objects. These knobs also both include units, in this case Hertz (Hz) next to their numerical values. Finally, the knobs are displayed, from left to right, in the order of "Gain," "LPF Cutoff," and "HPF Cutoff," which is congruent with our @displayorder attributes.

Upon exportation to Emote, the three knobs shown in the screenshot above will be displayed to the user. The "gain" knob is unchanged from the previous two tutorials, although this knob now technically controls two separate *gain~* objects, one for each channel. The "LPF Cutoff" and "HPF Cutoff" knobs control the cutoff frequency of low and highpass filters. As previously discussed, our signal routing in RNBO is set up such that the lowpass filter only affects input channel 1, and the highpass filter only affects input channel 2.

6.4.5. RNBO Compressor Tutorial #1 – Simple Compressor

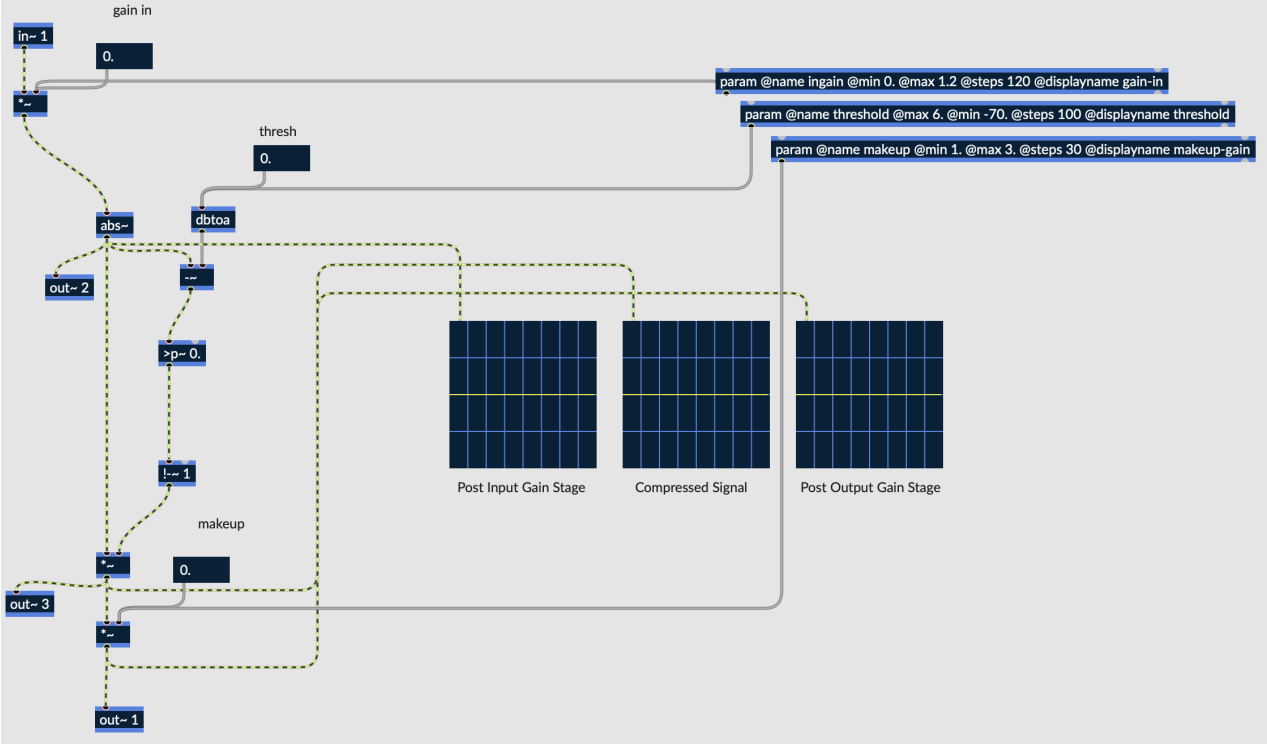


Fig. 6.83 **Simple dynamics compressor RNBO patch**

This example will show how a basic dynamics compressor can be built for the H9000 using RNBO and gen~. In part 2, we'll add compression ratio, as well as attack and release controls.

Download the maxpat file here: Simple Compressor.

The compressor has three main parts: input gain stage, compression stage, and output gain stage.

1. Input Gain Stage

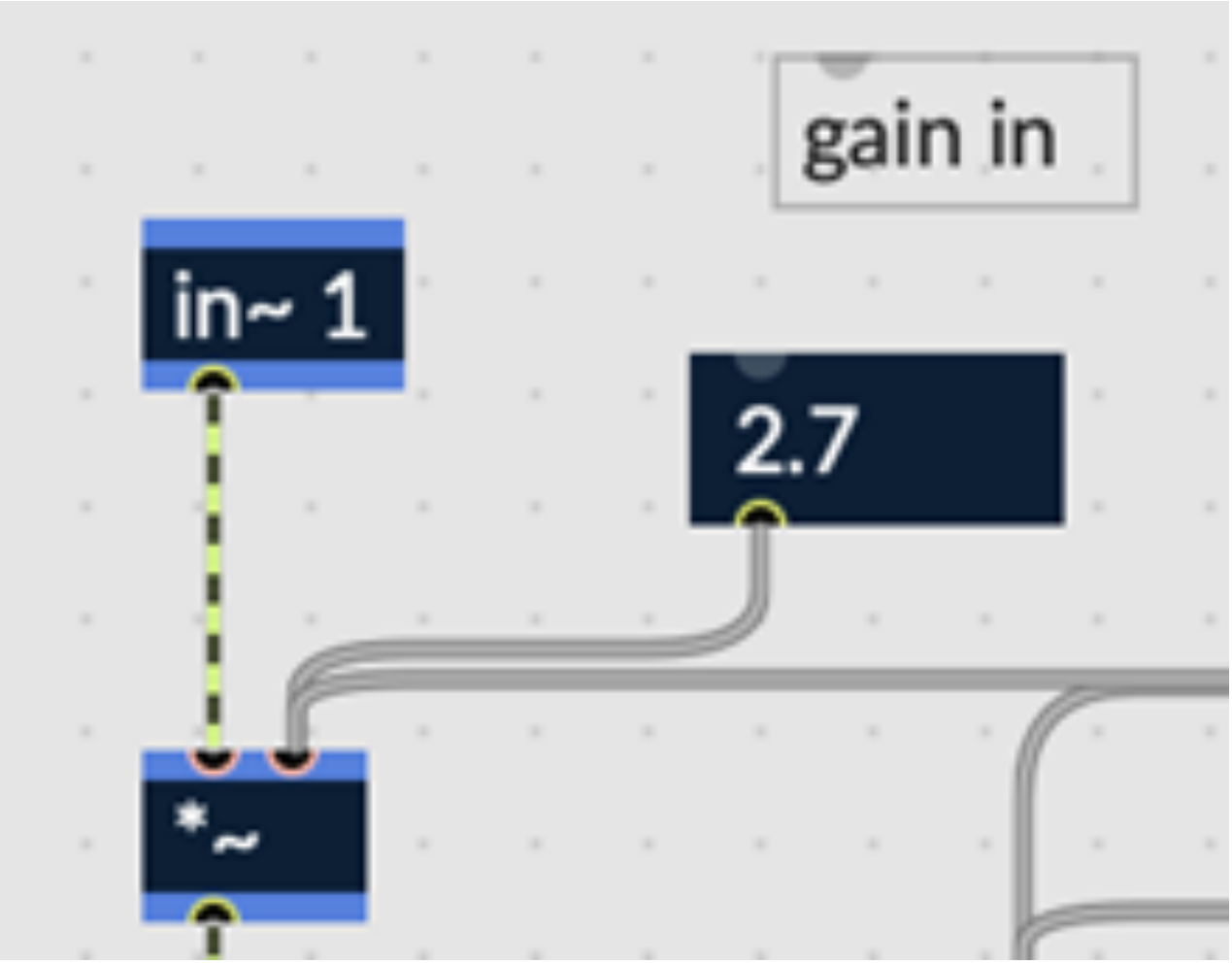


Fig. 6.84 **Input stage gain scaling**

This section simply determines the input level to the compression stage. Here linear gain scaling is used (1 = unity gain, the value is high because the soundfile in this case is quiet). You could control this with db gain using the *dbtoa* object.

1. Compression Stage

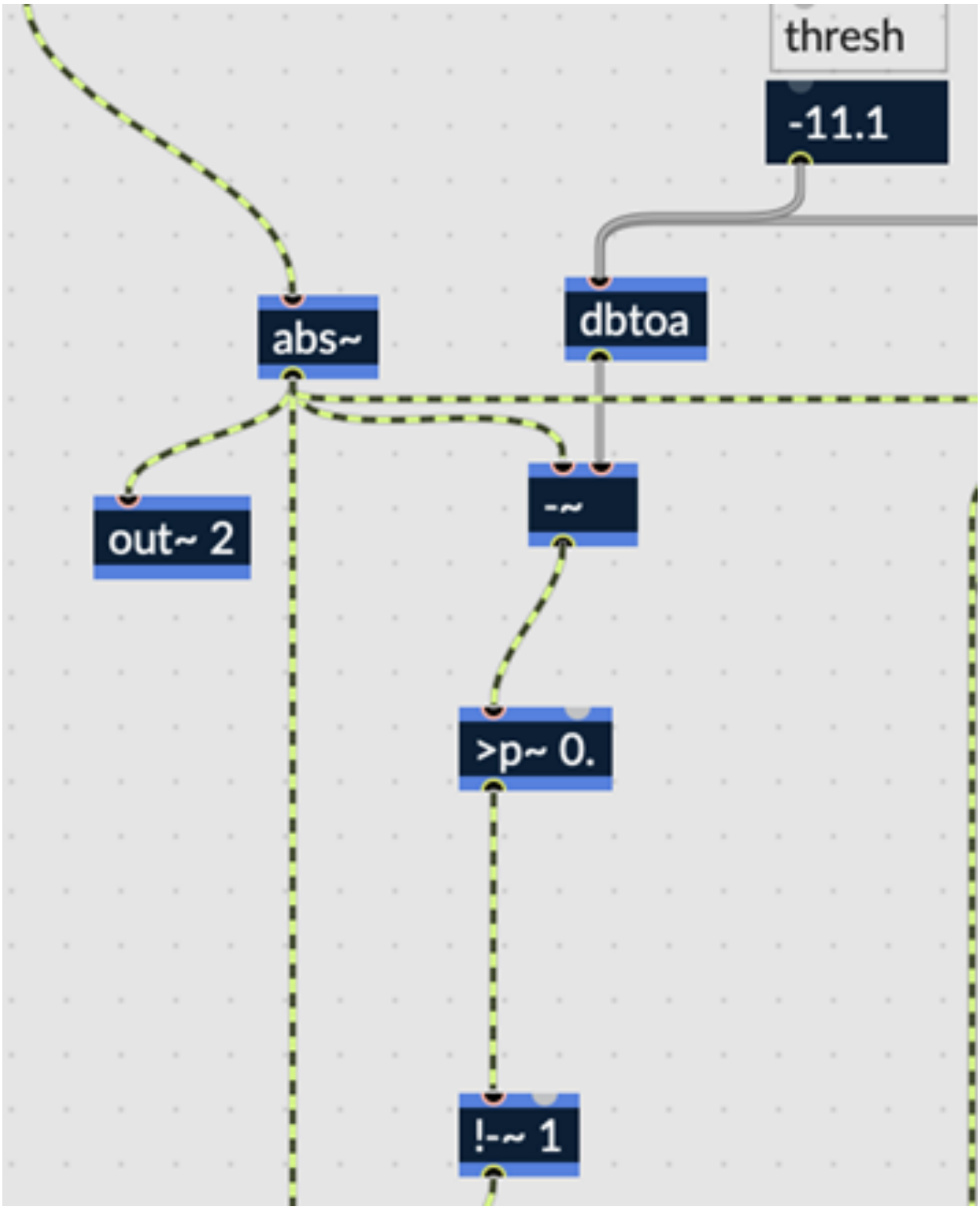


Fig. 6.85 **Compression stage**

There are two parts to the compression stage—gain detection and gain reduction. To detect the gain of the incoming signal, we first take the absolute value of the audio signal. This converts the gain of the input to a positive number regardless of its cycle position. Then, we subtract the threshold (converted to linear amplitude) from the signal amplitude. If the signal is above the threshold, the difference will be positive. If it is below the threshold, the difference will be negative. This value is the overshoot. [>*p~ 0.*] detects passes the overshoot value if it is greater than 0. Once passed, it is subtracted from 1. (unity gain) to give us our gain reduction factor. Finally, we multiply the original audio signal by the gain reduction factor. This gives us our compressed audio signal.

1. Output Gain Stage

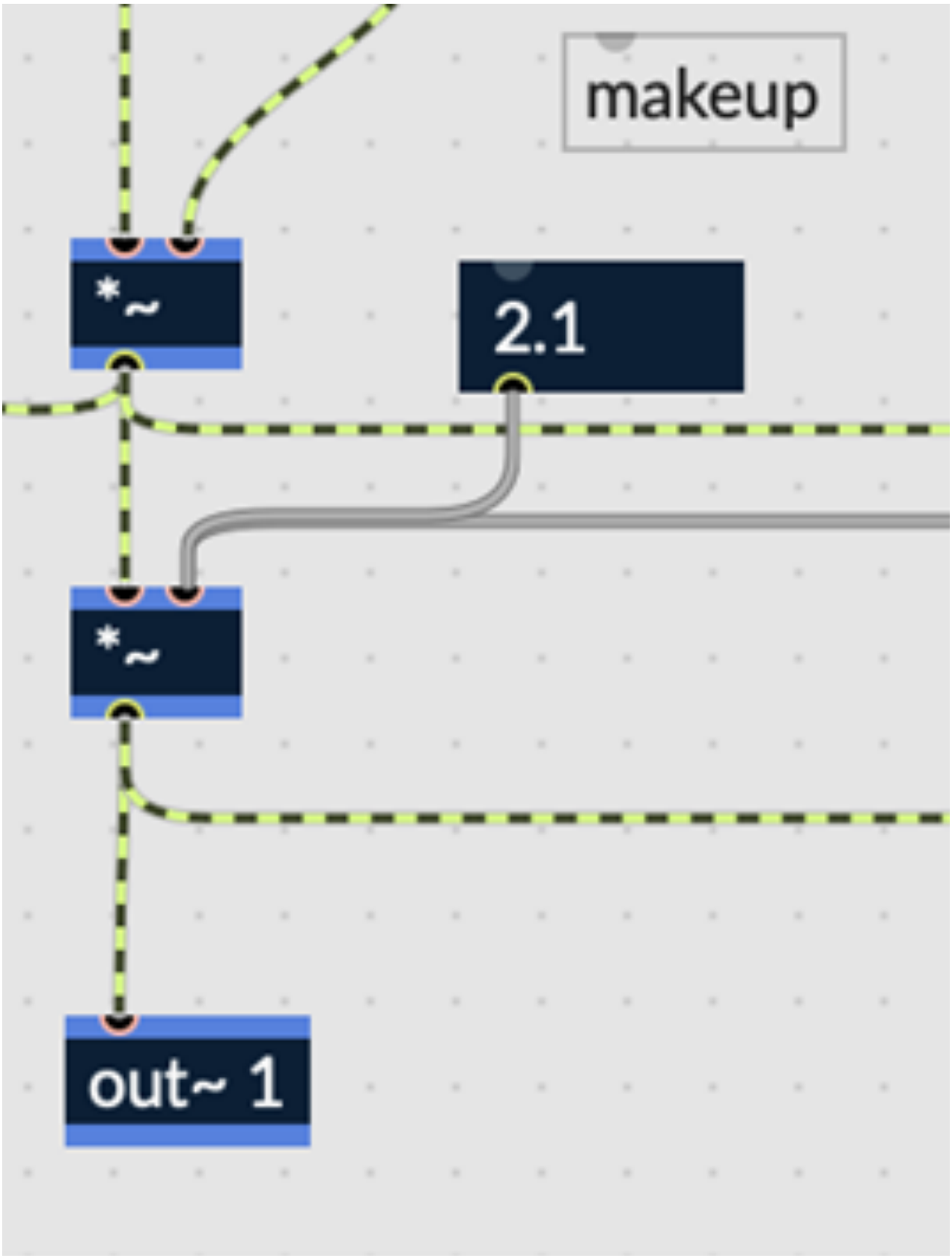

makeup
*~
2.1
*~
out~ 1

Fig. 6.86 **Output stage with make-up gain**

Lastly, we scale the compressed audio signal with a makeup gain factor. This allows us to compensate for the overall level decrease caused by the compression stage.

Fig. 6.87 **All *param* obects used by the patch, displayed as three knobs in Emote**

These *param* objects in RNBO allow us to control the compressor parameters in Emote after it is exported to the H9000. *Params* will appear as basic knobs in Emote (currently only knobs are supported when exporting RNBO patches to the H9000). The @min and @max arguments define the minimum and maximum values of the Emote control, @steps defines the number of steps between the min and max values, and @displayname sets what text will appear with the knob in Emote.

Fig. 6.88 **UI displayed in Emote, including three knobs corresponding to three *param* objects**

We now have a basic compressor designed in RNBO that can be exported to our FX chains on the H9000. You could use this as a simple, low DSP cost way to catch peaks when running very dynamic algorithms on the H9000, or use it as a starting point to design your own dynamics effects!

In parts 2-3, we'll look at some ways to upgrade our simple compressor.

6.4.6. RNBO Compressor Tutorial #2 – Ratio/Attack/Release

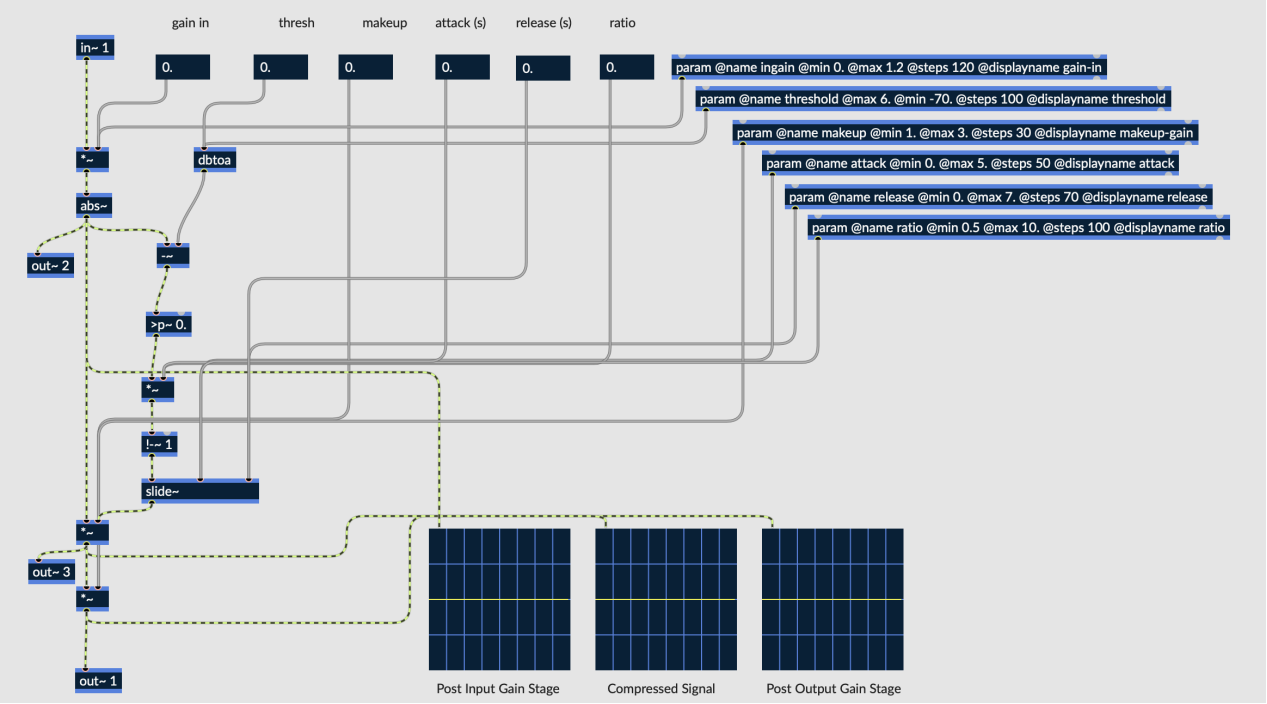


Fig. 6.89 **Dynamics compressor with ratio, attack, and release controls**

In this tutorial, we'll build on our simple compressor patch to include ratio, attack, and release controls.

The compression ratio allows us to scale the gain reduction about to change how much attenuation will be triggered by a given amount of threshold overshoot. To do this, we place a *~ object after >*p*~. For attack and release, we use the *slide~* object, which generates a logarithmically smoothed ramp between the incoming values with adjustable ramp up and down times. The ramp up time is our attack control, and the ramp down is release.

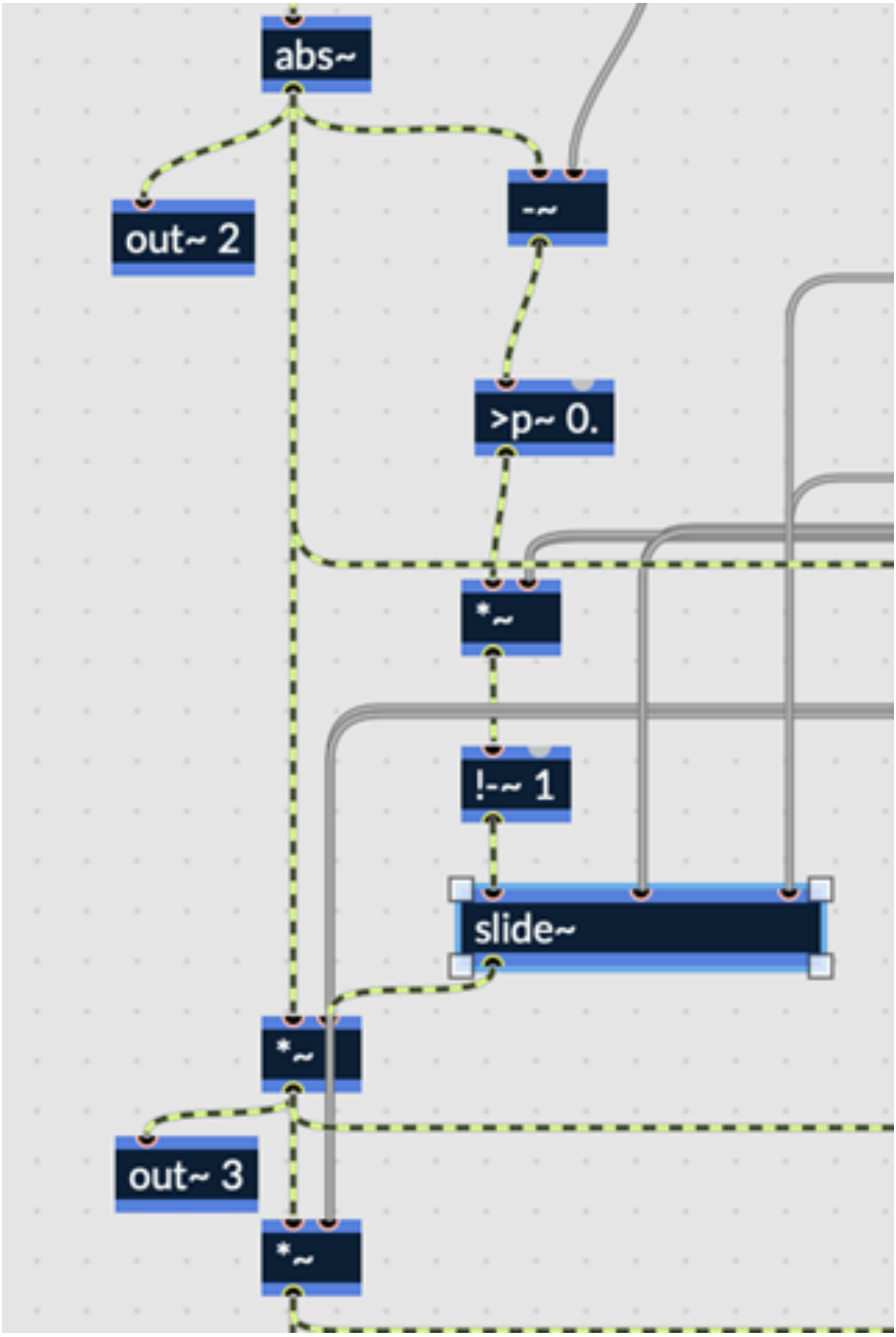

abs~
out~ 2
-~
>p~ 0.
*~
!-~ 1
slide~
*~
out~ 3
*~

Fig. 6.90 **RNBO signal structure for creating ratio, attack, and release controls**

And that's it! These controls allow us to fine tune our compression settings and achieve a smoother, less harsh compressed sound.

Below, you can see the compressor controls in Emote.

Fig. 6.91 **UI displayed in Emote, now including attack, release, and ratio controls**

6.4.7. RNBO Compressor Tutorial #3 – The RNBOmnipressor

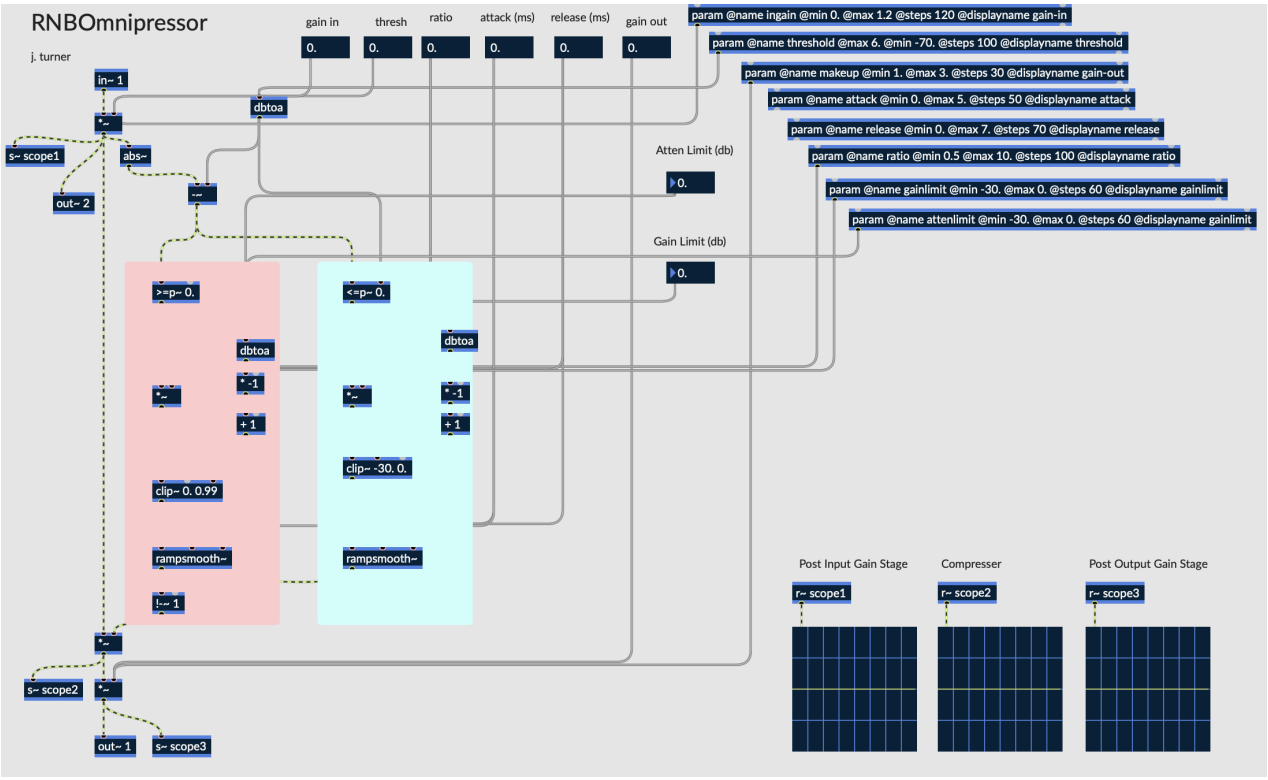


Fig. 6.92 **Dynamics compressor patch recreating Eventide Omnipressor**

For part 3 of our RNBO compressor, we're recreating the legendary Eventide Omnipressor. The Omnipressor is famous for its unique simultaneous compression and expansion functions with adjustable gain and attenuation limit controls. While tweaking its settings can yield many interesting (and sometimes bizarre!) gain curves, it's not too difficult to program in MAX/RNBO! In this example, the design has been expanded upon from the previous tutorial.

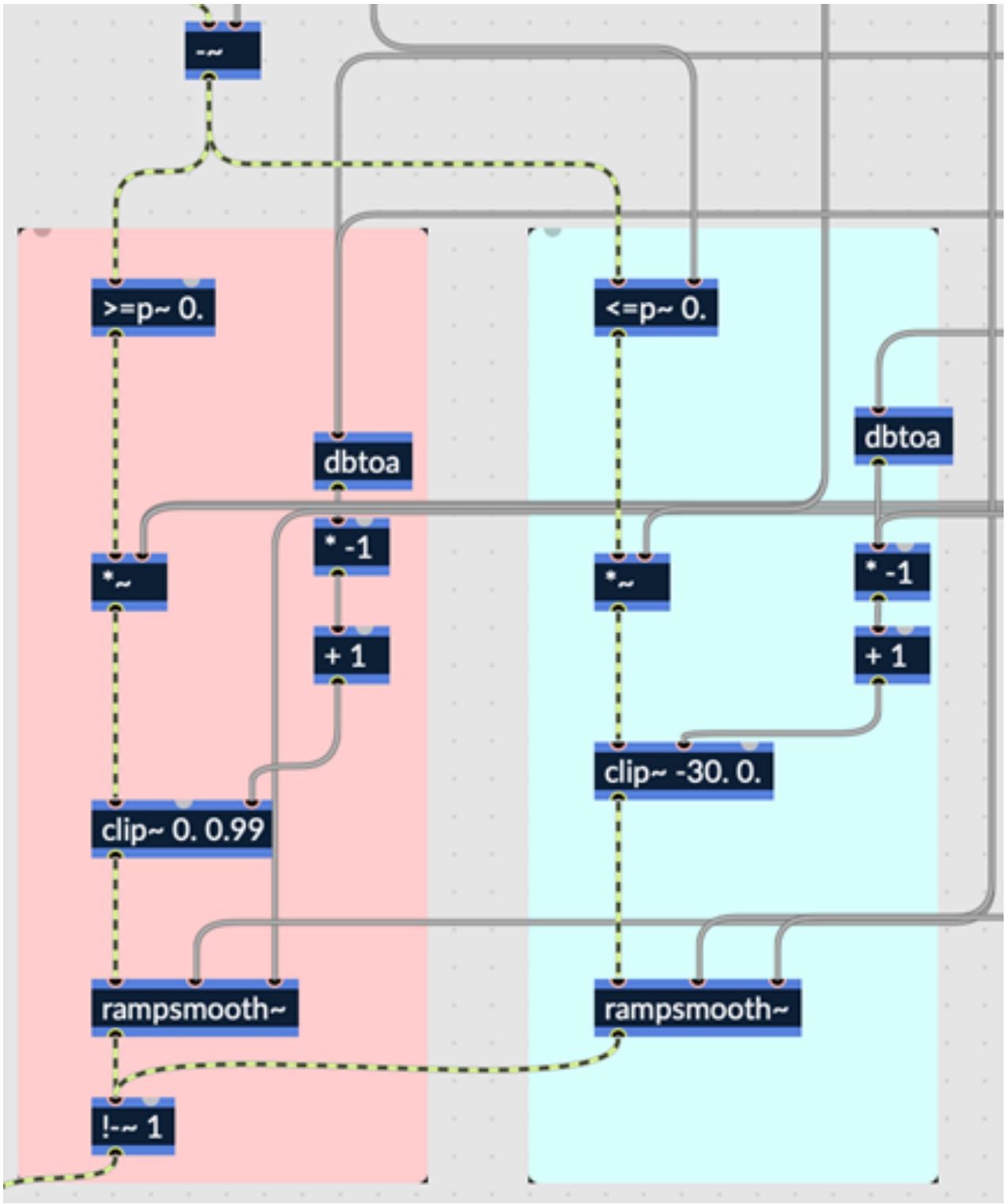


Fig. 6.93 **Compression (left) and expansion (right) algorithms, running side by side**

In this patch, we have compression and expansion algorithms running side by side. The compression algorithm (on the left) should look familiar—when the overshoot (the difference between the current amplitude and the threshold) is positive, that value is passed into the gain reduction section. Then, it is multiplied by the ratio.

The same thing happens on the expansion side, but in reverse: When the overshoot is negative (the signal is below the threshold), it is passed and scaled by the ratio. Because both processes run at the same time, a signal that is fluctuating above and below the threshold will be both compressed downward when it's above and expanded upward when it is below.

The *clip~* object allows us to create the gain and attenuation limit controls on the Omnipressor. These controls set limits on how much compression and expansion can happen regardless of the threshold or ratio. We use the *clip~* object for this, which only allows values to pass within a specified range. By changing the upper and lower limits on the compression and expansion sections, respectively, we can control the attenuation and gain limits. Note: the compression factor is negative and the expansion factor is positive here because they are eventually reverse subtracted from 1.

Finally, the *rampsmooth~* object allows us to control attack and release times. In this case, having one for each section would allow us to set different attack and release times for compression and expansion. After that, the compression and expansion factors are applied to the input signal, giving us omnipressed audio! Below, you can see the controls in Emote after the patch is exported to the H9000.

Fig. 6.94 **UI displayed in Emote, now including expansion controls**

In these tutorials we've seen how a simple RNBO effect can be turned into something more complex and musical. These patches can be exported to your H9000 and controlled via Emote, where you can use them alongside your favorite algorithms!

6.4.8. RNBO Delay Tutorial #1 - Stereo Delay

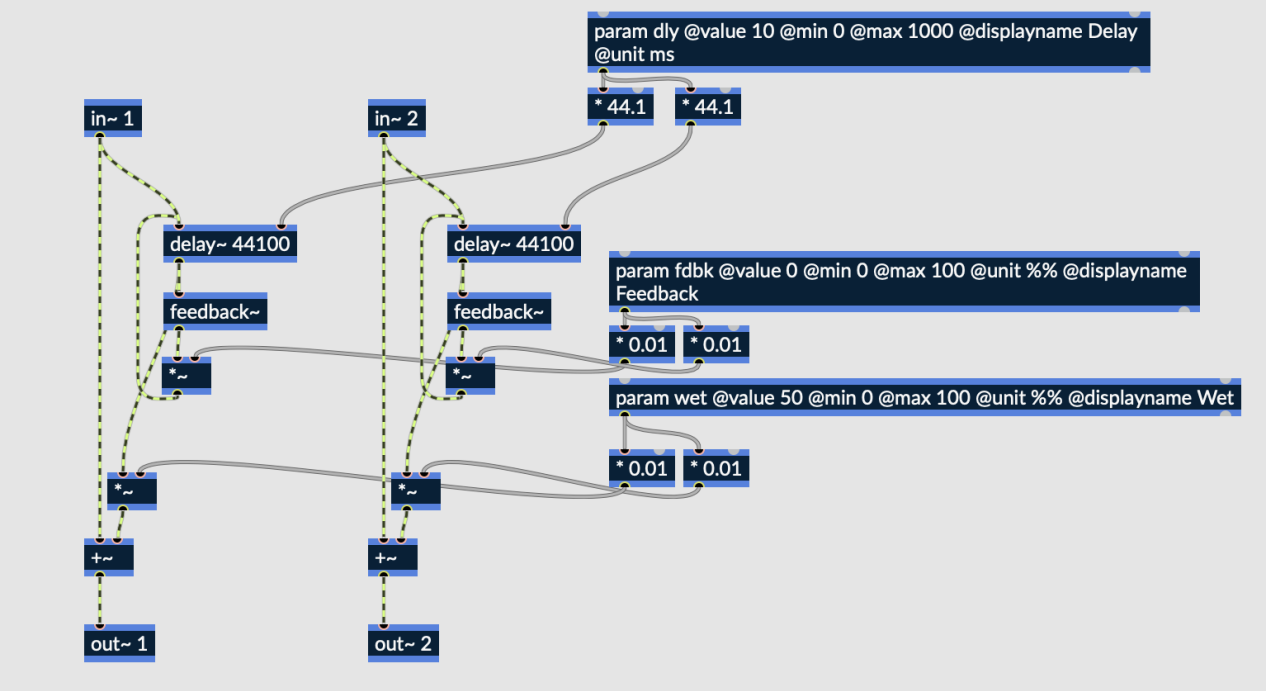


Fig. 6.95 **Stereo delay patch in RNBO**

The above patch is a stereo delay with feedback. Despite its relative simplicity, this patch can be easily upgraded to achieve more complex and dynamic effects. This tutorial is the first in a series of four, building a foundation upon which we will expand with additional parameters, modulation, and cross feeding.

Download the maxpat file here: Stereo Delay.

6.4.8.1. Delay Structure

The most important element of this patch is the *delay~* object, which delays an incoming signal by some number of samples. *Delay~* requires a maximum size argument, which has been set to 44,100 samples (1 second) in this case.

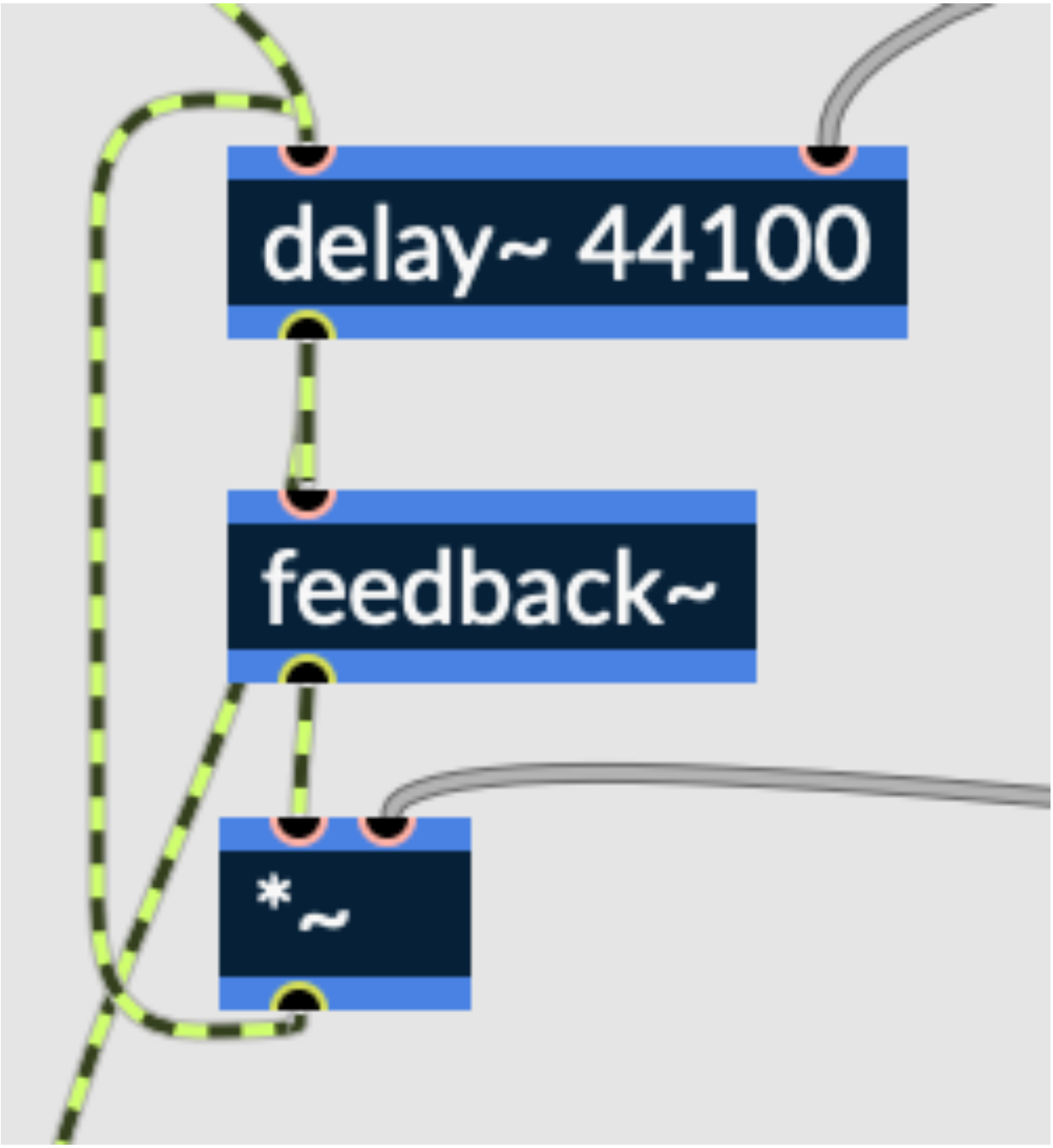


Fig. 6.96 **Delay feedback loop**

To create a feedback loop for our delay, the *delay~* output cannot simply be routed back to its input. The *feedback~* object must be used, which provides a single sample buffer delay. The output of the *feedback~* object can then be scaled and routed back to the *delay~* input.

6.4.8.2. Delay Parameters

This patch uses delay time (dly), feedback (fdbk), and a wet signal control (wet) as its three main parameters. You may have noticed that each *param* output is scaled by some constant coefficient. This is done so that the user may have parameters displayed to them in units which they are accustomed to. For example, RNBO's *delay~* object measures its delay in **samples**, as opposed to milliseconds.

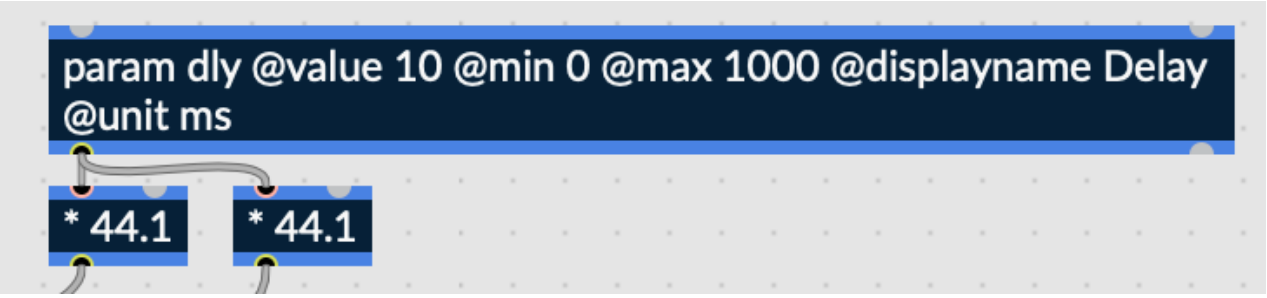


Fig. 6.97 **Delay *param* object, scaled by 44.1 on output**

By multiplying the param output by 44.1, milliseconds are converted to samples which can be processed by the *delay~* object. (Be aware that this scaling factor is dependent on the sampling rate which you are working with).

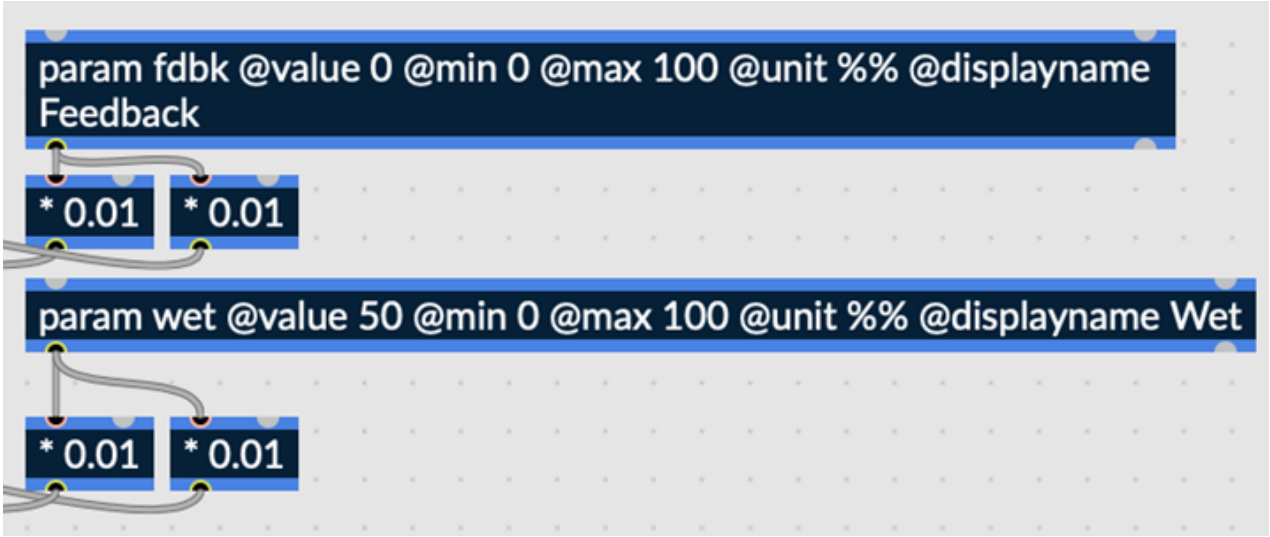


Fig. 6.98 **Feedback and wet control *param* objects, both scaled by 0.01 on output**

Similarly, the feedback and wet signal controls are scaled by 0.01. Both the feedback and wet signal controls work by sending a value to the *~ object, which performs multiplication on audio signals. For both the wet and feedback controls the scaling factor should range from 0 to 1, but this is not very intuitive for the end user. Hence, the wet and feedback param objects output between 0 and 100 and are scaled by a factor of 0.01.

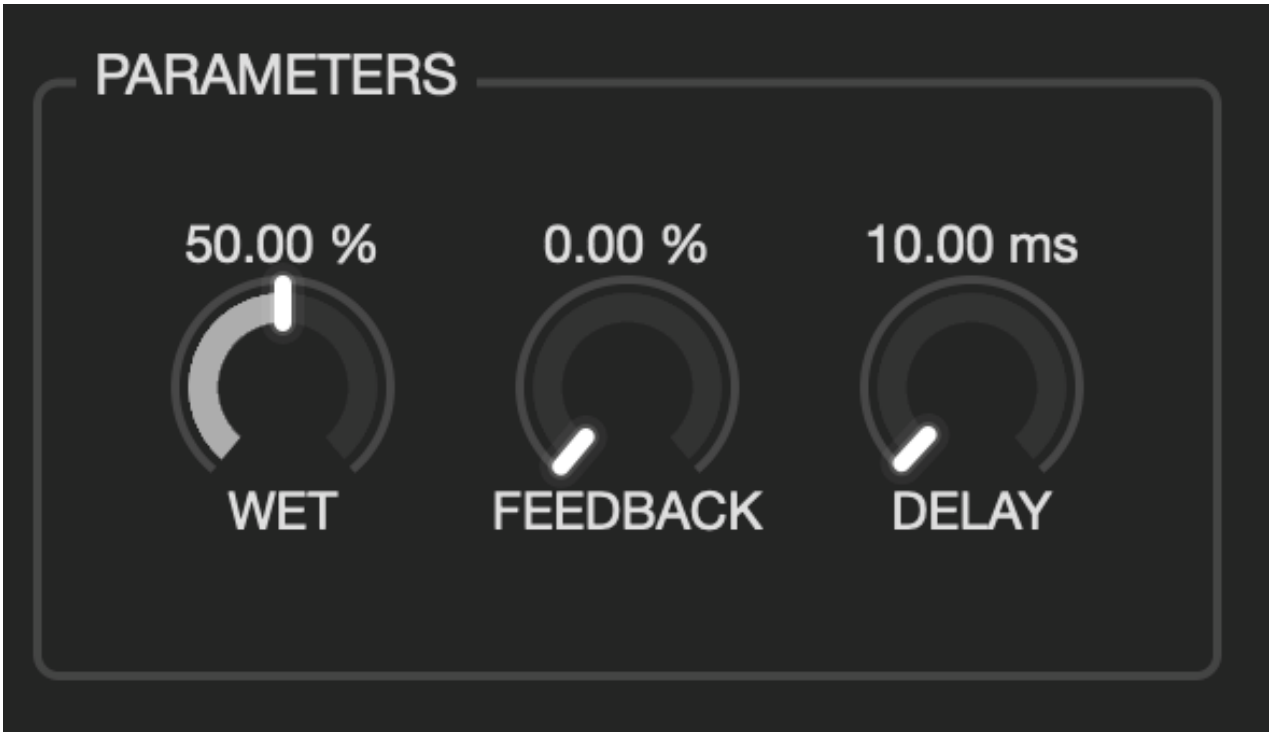


Fig. 6.99 **UI displayed in Emote**

Upon exporting to Emote, the above menu will be displayed to the user. All parameters are displayed intuitively and in units that most users will be used to. Wet and feedback controls are measured as a percentage, while delay is measured in milliseconds. To make this patch slightly more complex, one may consider separating delay and feedback controls by left and right channels. This patch is also very easily upgradable to a flanger style modulated delay. The next tutorial will explore both upgrades.

6.4.9. RNBO Delay Tutorial #2 - Stereo Flanger

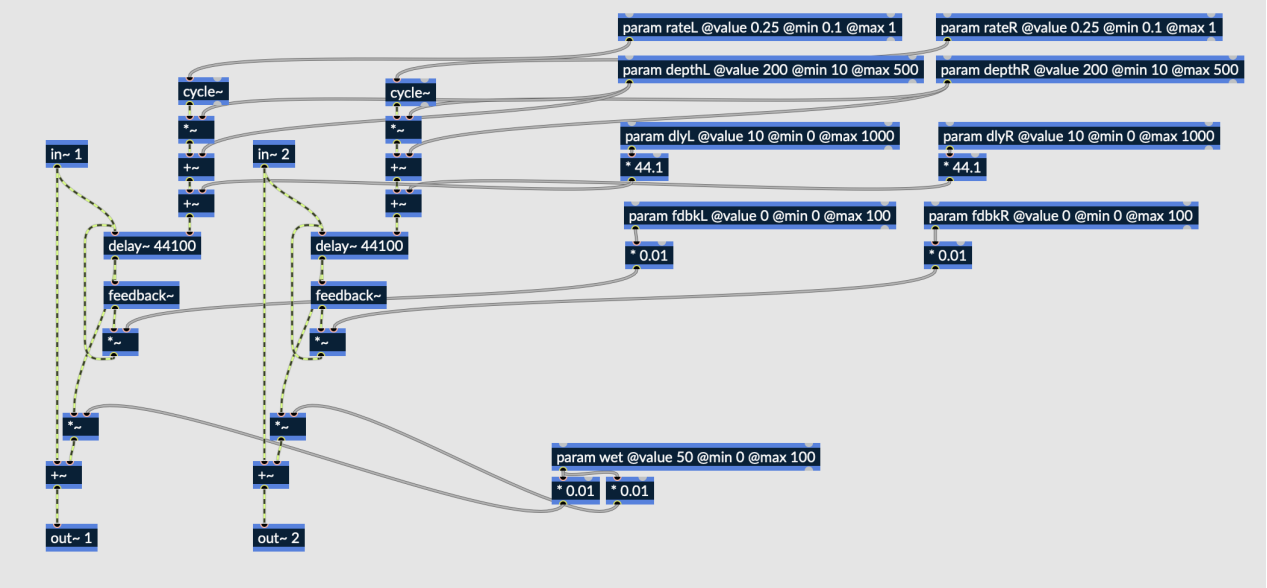


Fig. 6.100 **Stereo flanger patch in RNBO**

The above patch expands upon the Stereo Delay patch from the last tutorial. Not only are the parameters present in the last patch split into individual controls for the left and right channels, the delay lines are now modulated to create a flanging effect. The addition of modulation necessitates the introduction of the "rate" and "depth" parameters, both of which can be individually controlled for each channel.

Download the maxpat file here: Stereo Flanger.

6.4.9.1. Modulated Delay Structure

The delay lines themselves are unchanged from the Stereo Delay patch, but the delay time input now varies overtime. The *cycle~* object acts as the main delay input, which outputs a sine wave. The frequency of the sine wave is controlled by the "rate" parameter. For the sine wave to behave exactly how we want it to, some basic operators are used.

Fig. 6.101 **Modulated delay structure**

1. The sine wave first has its amplitude scaled by constant factor, which is controlled by the “depth” parameter.

2. The amount of depth is then added to the sine wave output so that the output is always positive.

3. Finally, another constant is added to the sine wave, controlled by the “delay” parameter. This effectively acts as a delay offset, if the user desires an initial delay to the signal in addition to the flanging effect.

The output is ultimately multiplied by 44.1 for the same reason the delay param is multiplied by 44.1 (see Delay Tutorial #1 for more details).

Before performing these operations, the output of *cycle~* will look something like this:

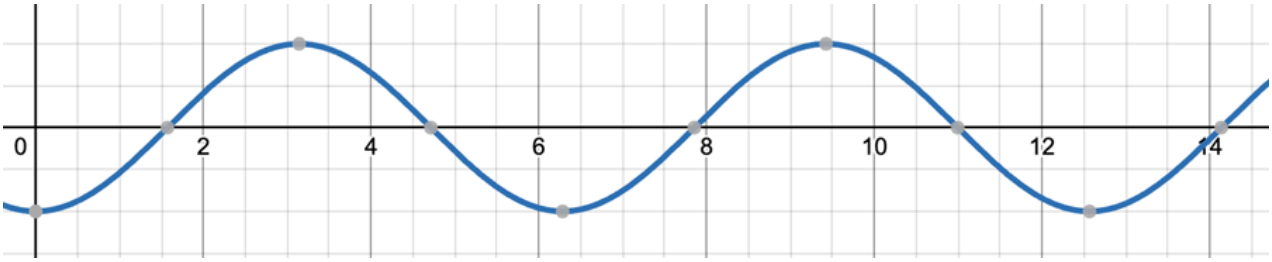


Fig. 6.102 **Graphical representation of *cycle~* output, before performing the above operations**

The result of these operators will cause the wave to look something like this:

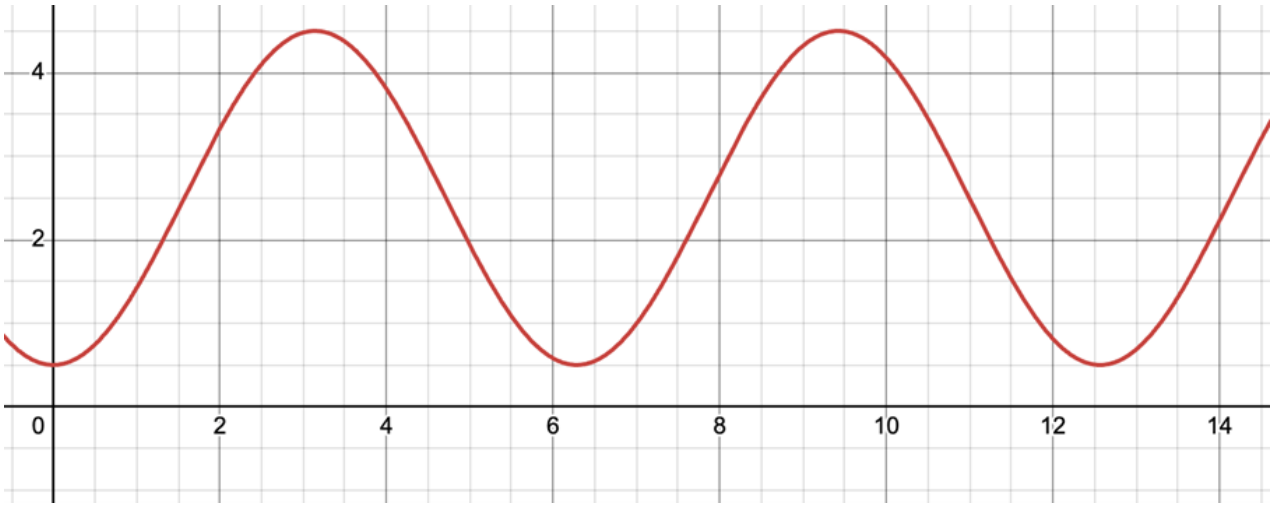


Fig. 6.103 **Graphical representation of *cycle~* output, after performing the above operations**

Operator #1 scales the wave's amplitude, while operator #2 vertically shifts the wave by the same amount that its amplitude was scaled. With only these two operations having been performed, the wave varies in output between 0 and double its amplitude. Finally, operator #3 provides an additional vertical shift. The result is a unipolar sine wave with some amount of vertical offset.

Recall that this sine wave represents the amount by which a signal is delayed as a function of time. The physical interpretation of this function, as previously discussed, is a delay line with a constantly changing delay time.

6.4.9.2. Additional Delay Parameters

Two new parameters, depth and rate have been added to this patch. Also, all parameters in the last patch have been doubled to allow independent parameter control over the left and right channels. We have already explored in detail how these parameters are utilized within the patching environment, but it may be helpful to describe these parameters more generally:

- Depth – the amount of delay modulation, measured in milliseconds; in other words, the amount by which the delay is allowed to fluctuate
- Rate – the rate at which delay fluctuates, measured in Hertz; the greater the rate and depth, the more intense the flanging effect tends to be.

Although the “delay” parameter was present in the last patch, it now behaves somewhat differently. Instead of acting as the time between the onset of the original signal and the delayed signal, it acts as the time between the onset of the original signal and the flanged signal. The difference is subtle, but worth acknowledging.

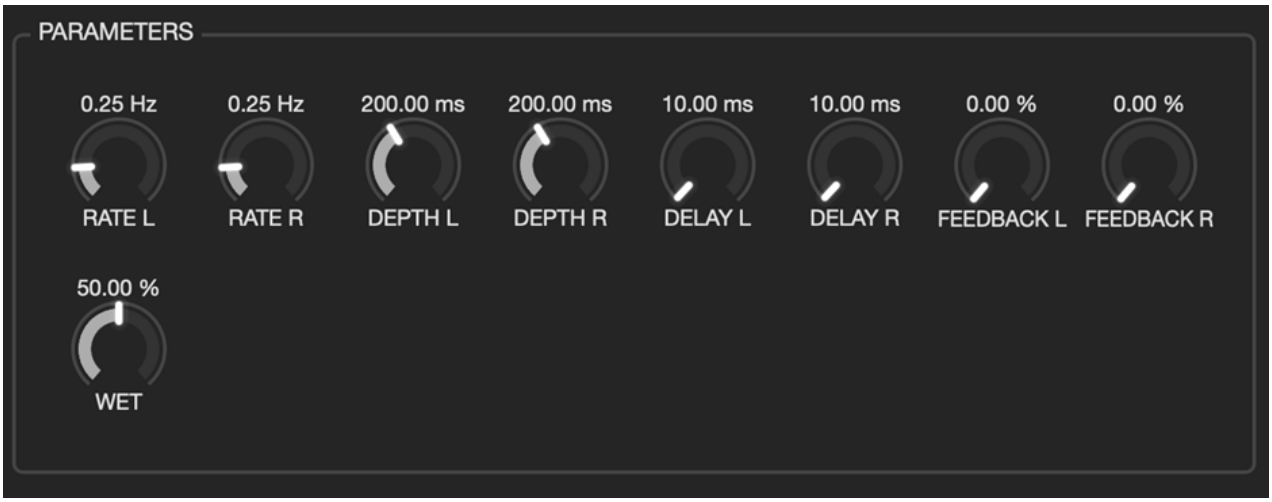


Fig. 6.104 **UI displayed in Emote**

Upon exporting to Emote, the above menu will be displayed to the user. With minimal additional patching, the original Stereo Delay effect has been upgraded to a Stereo Flanger with independent control of parameters for each channel.

6.4.10. RNBO Delay Tutorial #3 – Stereo Flanger with Crossfeed

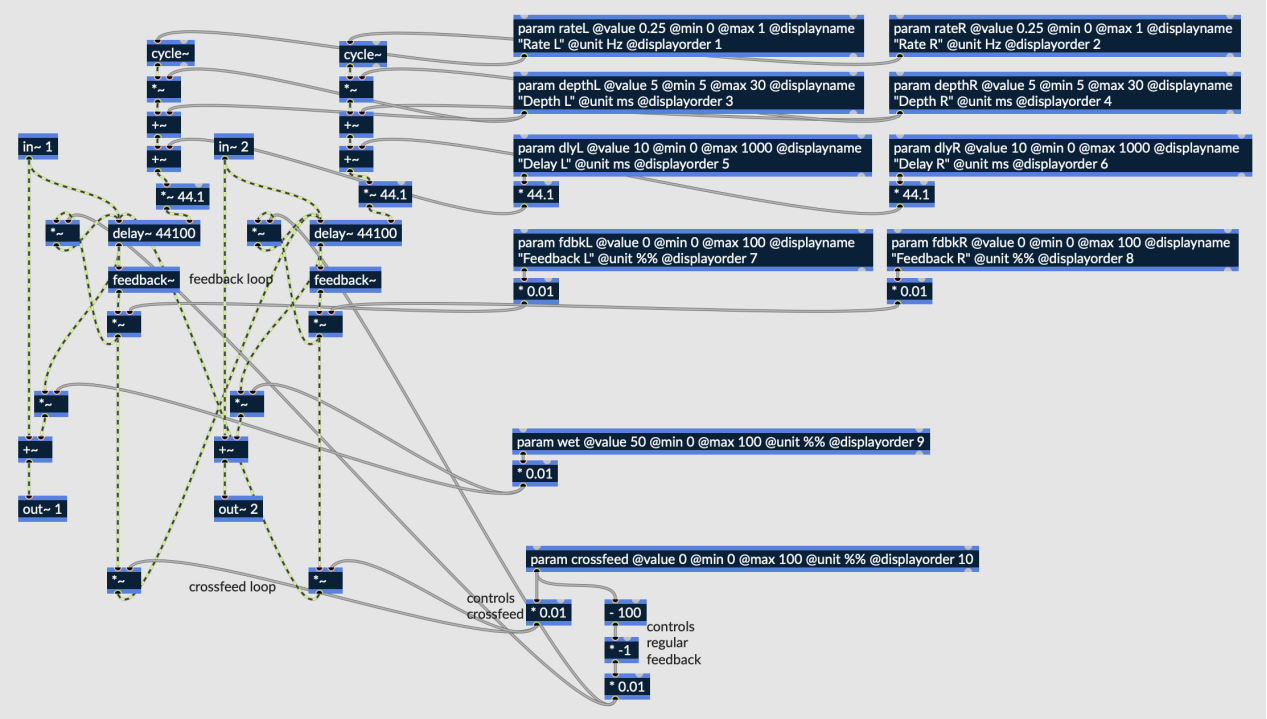


Fig. 6.105 **Stereo flanger patch with crossfeed in RNBO**

The above patch expands once more on the previous patch, adding crossfeed functionality to the flanger. While this tutorial only adds one parameter to the overall effect, it is a slightly more complex use of the param object than has been explored previously.

Download the maxpat file here: Stereo Flanger with Crossfeed.

6.4.10.1. Crossfeed

When working with stereo delay effects, crossfeed refers to the process of routing the output of a delay line to the input of the opposite channel's delay line. This differs slightly from a typical delay line structure, which looks like this:

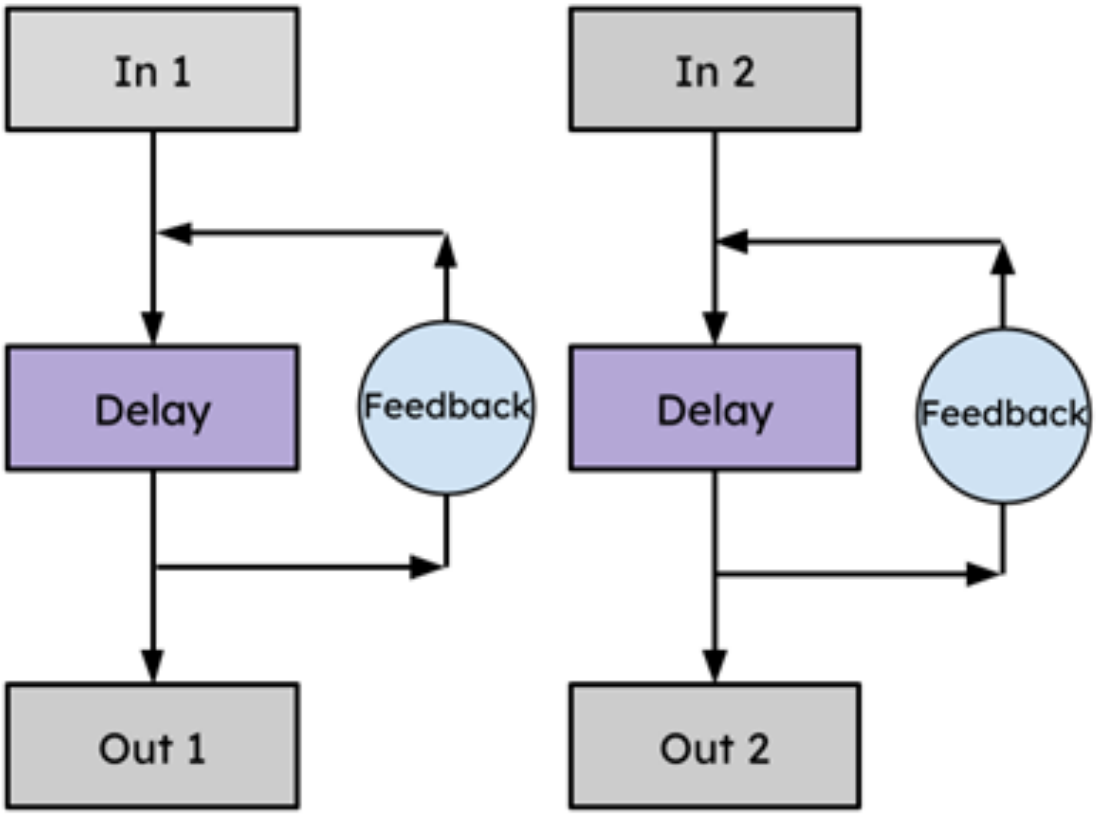


Fig. 6.106 **Block diagram demonstrating a stereo delay with feedback**

Both delay lines act independently of each other, with their own feedback loops. Crossfeed effectively allows the delay lines in a stereo delay to interact with one another by crossing signal into each other's feedback loops (hence crossfeed). A stereo delay with crossfeed can be represented by the following:

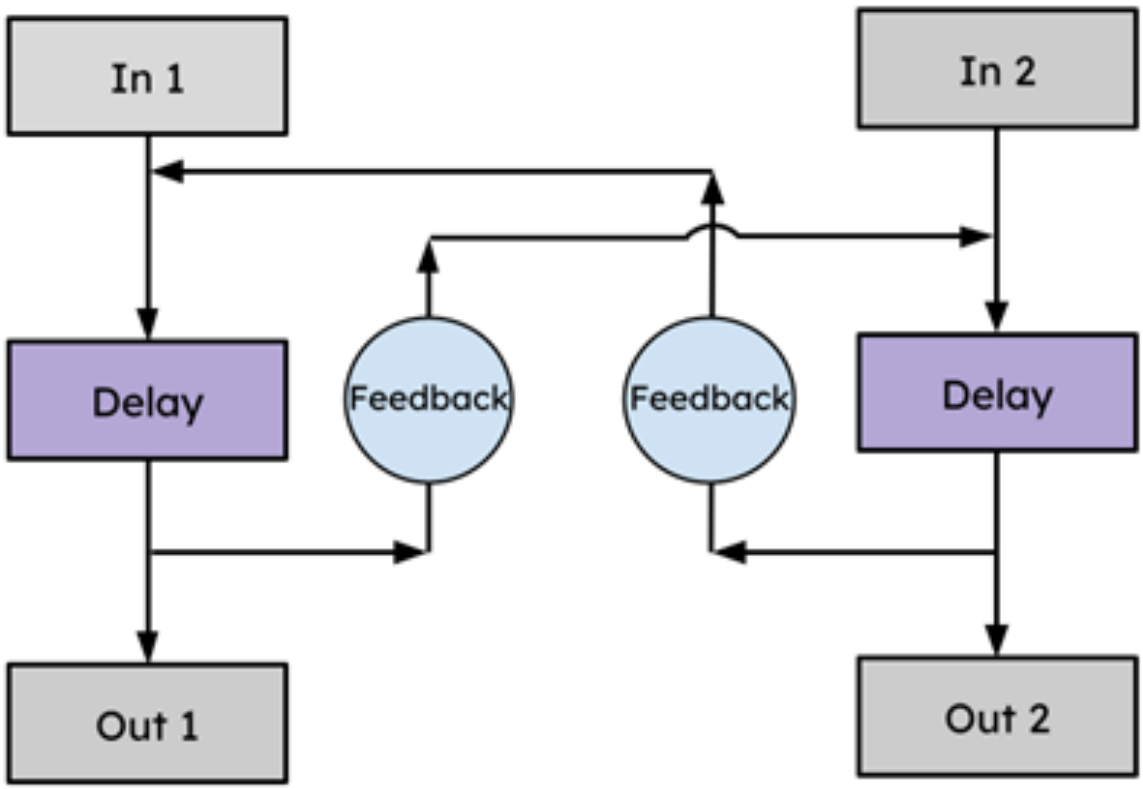


Fig. 6.107 **Block diagram demonstrating a stereo delay with crossfeed**

Crossing feedback lines helps create rich stereo textures that cannot be achieved with a typical delay line structure. When also used in conjunction with modulation, the possibilities may seem endless.

6.4.10.2. Implementing Crossfeed in RNBO

The process for implementing crossfeed in a RNBO patch mostly involves the use of *~ objects, and the addition of a single *param* object. For the purposes of this patch, crossfeed will be a singular parameter which controls both the amount of left channel delay which is crossed to the right, and the amount of right channel delay which is crossed to the left (Independent control of crossfeed for each channel can be implemented very similarly).

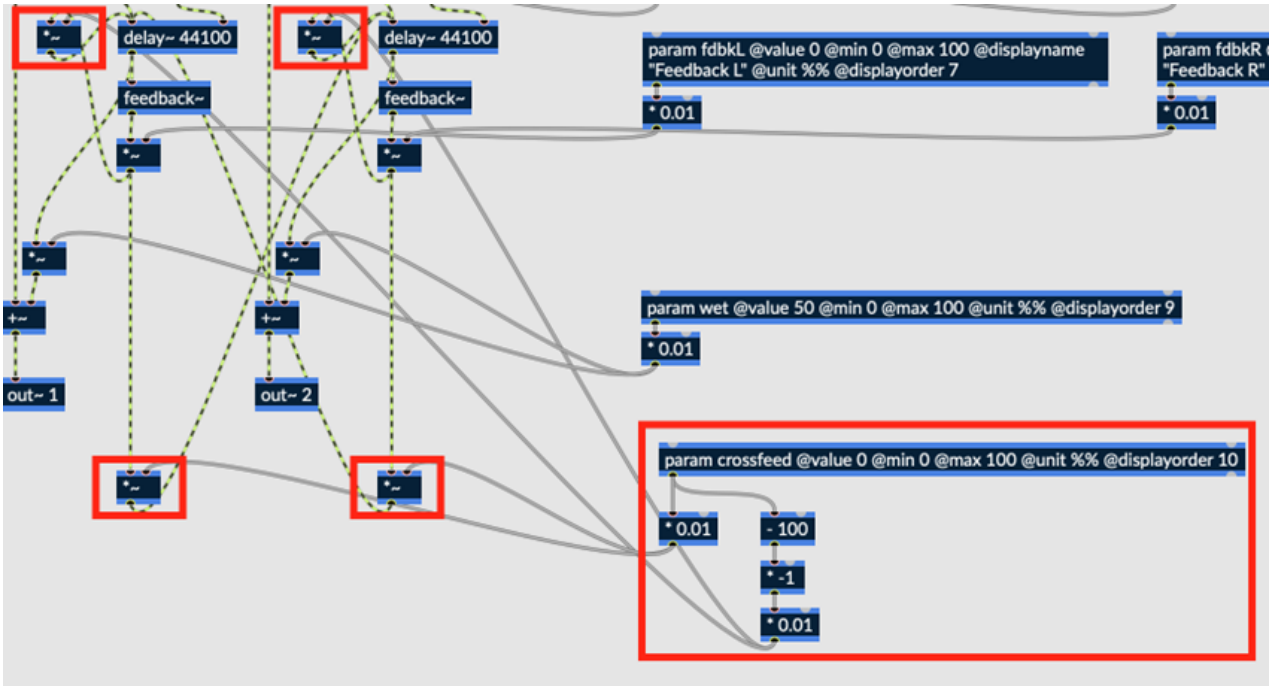


Fig. 6.108 **Crossfeed *param* controls four *~ objects simultaneously, maintaining a balance between feedback and crossed feedback**

The crossfeed *param* controls four separate *~ objects. The two *~ objects on the bottom control the amount of crossfeed, while the two on top control the amount of regular feedback. Let's look a little closer at the *param* object itself.

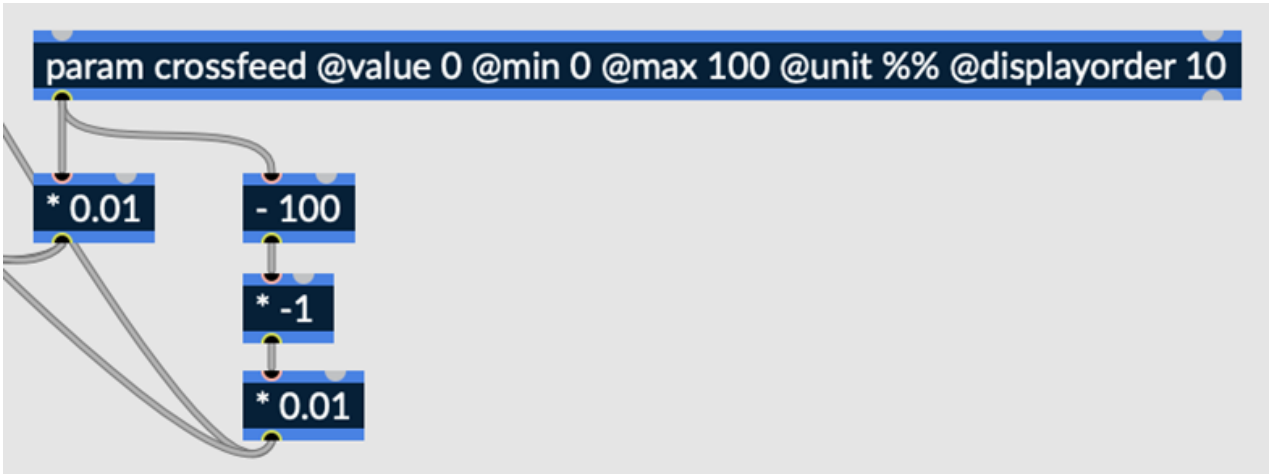


Fig. 6.109 **Crossfeed *param* object**

Some basic operations are applied to the *param* immediately upon output. This is to ensure that the percentage of crossfeed and the percentage of regular feedback always sum to 100%. Similarly to the wet and feedback controls introduced in the first Delay Tutorial, the crossfeed output is scaled by 0.01 so that the user may control crossfeed in terms of percentages.

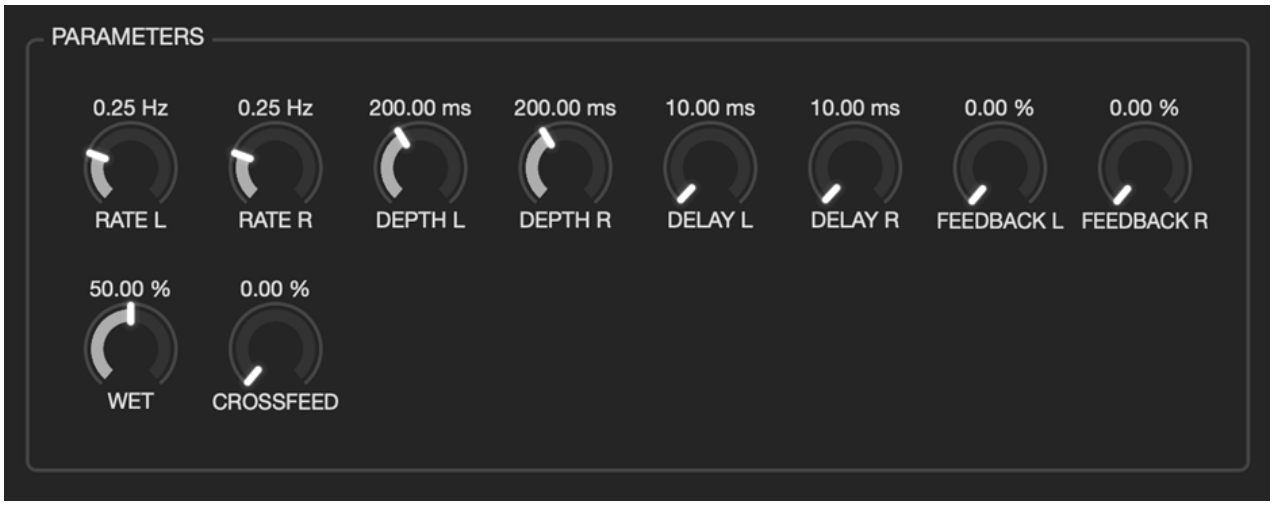


Fig. 6.110 **UI shown in Emote**

Upon exporting to Emote, the above menu will be displayed to the user. With the addition of one parameter, the basic Stereo Flanger has been upgraded beyond the essentials. When working in RNBO, building basic, tried and tested effects is a great way to become acquainted with the patching environment and hone your Max/MSP skills. With some practice, these basic effects can be updated and modified with more complex functionality. The only limit is your imagination.

6.4.11. RNBO Delay Tutorial #4 – Multiple Modulation Sources

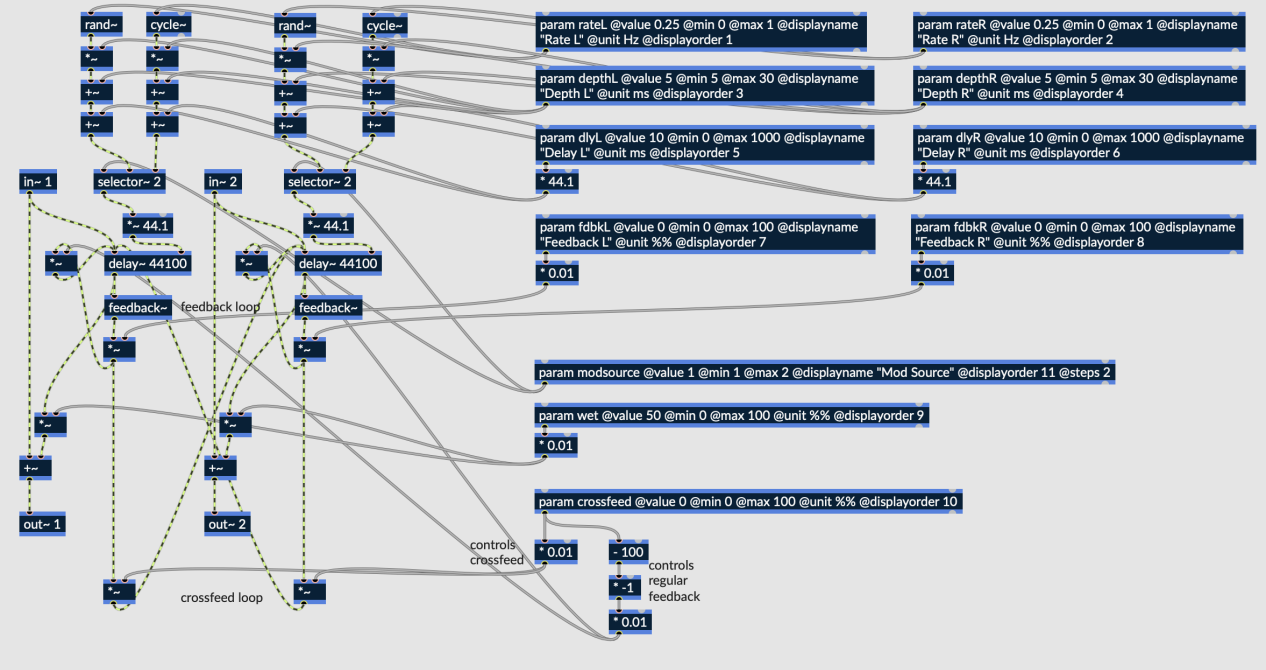


Fig. 6.111 **Crossfed stereo flanger with multiple modulation sources**

The above patch is the final in the Delay Tutorials series and expands on the previous patch by adding an additional modulation source, and a parameter which allows the user to select their desired modulation source.

Download the maxpat file here: Multiple Modulation Sources.

6.4.11.1. Selector~

The process of adding another modulation source is quite simple. In this case, the *rand~* object is used, which is a band limited random signal generator. The operations performed on the random signal, and their associated parameters, are the same as those performed on the sine wave signal generated by *cycle~*. In fact, the same param objects are used to control both mod sources.

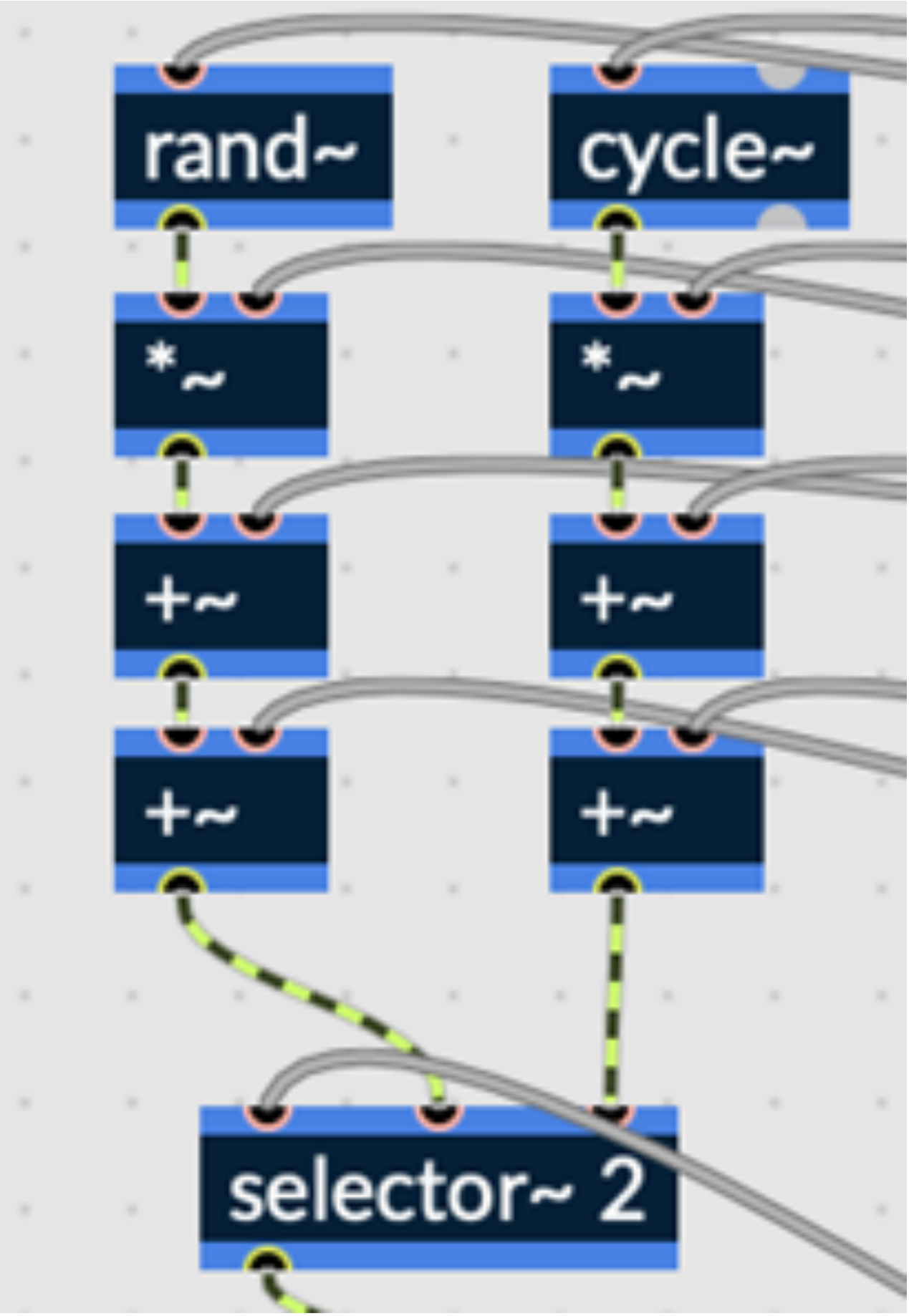

rand~
cycle~
*~
*~
+~
+~
+~
+~
selector~ 2

Fig. 6.112 ***Rand~* and *Cycle* objects both being used as modulation sources, toggleable via a *Selector~***

Switching between modulation sources, however, requires the addition of *selector~* objects. *Selector~* accepts inputs from several sources, assigning one input at a time to the output. A new *param* object, labeled "modsource," sends a value to the leftmost inlet of *selector~*, which selects which input is to be passed to the output. A value of 1 passes the random signal, while a value of 2 passes the sine wave.

Fig. 6.113 **UI displayed in Emote**

Upon exporting to Emote, the above menu will be displayed to the user. The addition of a modulation source has not significantly changed the algorithm, but now allows for a slightly different texture than was available with the regular Stereo Flanger. The *rand~* object creates a modulation effect more akin to a chorus.

6.4.12. Parameters and Creating UI

In its current form, RNBO's H9000 support does not allow for the same level of advanced UI creation that is possible in VSIG. While VSIG includes several unique UI modules and an intuitive menupage system, UI customization in RNBO occurs entirely within the [param] object. While this may seem like a limitation, the [param] object's extensive list of optional attributes makes it much more capable than one might expect at first glance.

Download the maxpat file here: Parameters and Creating UI.

6.4.12.1. Param Arguments/Attributes

When exporting any RNBO patch to the H9000, all *[param]* objects are represented with knobs, or switches in some cases, by default (the switch is a special case, which will be explored later in its own section). There is currently no way to have *[param]* objects represented by other UI objects, such as faders or text boxes. Despite this limitation, *[param]* arguments/attributes allow for great customization over how UI objects behave in Emote.

Arguments – arguments are required when instantiating an object, and must be entered in the proper order:

- Name – name of the parameter
- Value – set initial value

Arguments may be specified with the argument name followed by an @ sign, but this is not required. Either of the below examples is a valid instantiation of the *[param]* object.

Fig. 6.114 **Two valid instantations of a *param* object, using no optional attributes**

Attributes – like arguments, attributes tell the *[param]* object how to behave; unlike arguments, however, they are not required and may be specified in any order:

- Min/mininum – minimum value

- Max/maximum – maximum value
- Steps – divide the output into a number of discrete steps
- Displayname – a more readable name for the parameter in Emote
- Displayorder – order in which parameters will show up in a list of parameters and in Emote; order can be numeric or symbolic ('first' and 'last')
- Unit – symbol to describe the unit of the parameter in an Emote

(The above attributes are not an exhaustive list of every [param] attribute; rather, they are the most important attributes for menu creation in Emote/H9000. For an exhaustive list, see the RNBO reference files.)

Let's dive a little deeper into a *[param]* object which utilizes all the attributes listed above. The following comes from Delay Tutorial #3 – Stereo Flanger with Crossfeed.

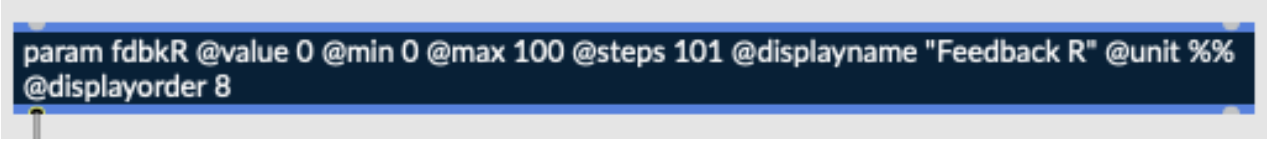


Fig. 6.115 **Param object utilizing several optional attributes**

This parameter controls the amount of delay feedback for the right channel. Looking into the attributes, we can predict how the corresponding knob should look in Emote:

- Initial knob position should be at 0 when loading the algorithm.
- Knob can access values as high as 100, and as low as 0.
- Knob can only access values in 101 discrete steps. In other words, output is limited to whole numbers.
- Knob will be labeled as "Feedback R," as opposed to the parameter name "fdbkR"
- Percent sign (%) will appear after the numeric value associated with the knob ("%%" is seen by H9000 as a single percent sign. VSIG syntax uses % sign elsewhere, and will not recognize a single % sign if specified as a unit)
- From left-right and top-bottom, knob will be the 8th

Upon exporting the patch to the H9000, the following is shown in Emote:

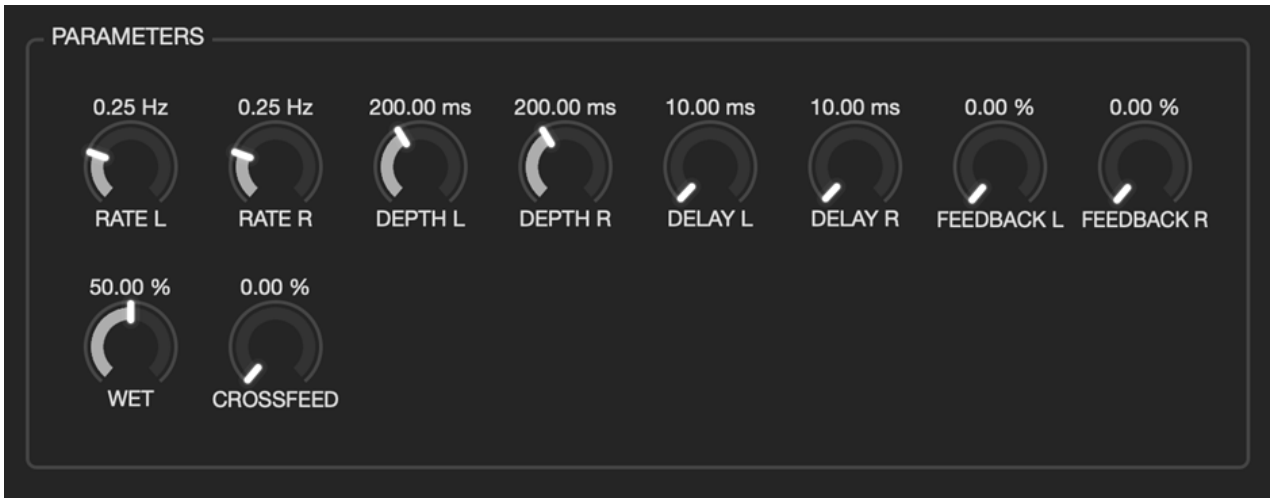


Fig. 6.116 **Notice the "Feedback R" knob, and how its attributes affect its properties and behaviors**

Through the use of attributes, we have created a more organized and user friendly UI than if no attributes were specified. While the "max" and "steps" attributes are not demonstrable in a screenshot, rest assured the "Feedback R" knob can access values only up to 100, in whole number increments.

6.4.12.2. Special Case - Switches

If and only if a *[param]* object is instantiated such that it can only access two different values, it will be displayed in Emote as a binary switch. Such an instantiation would look like so:

Fig. 6.117 **After exporting to Emote, the above *[param]* is displayed as a switch**

By limiting the values accessible by the *[param]* object to only 0 and 1, it will be displayed as a toggle switch in Emote:

Fig. 6.118 **UI object in Emote corresponding to *[param]* shown above, representing a toggle switch**

This is currently the only way to display a non-knob UI object in Emote using RNBO.

6.5. gen~

The H9000 has a C/C++ plugin API used in collaboration with Cycling '74 to allow gen~ patches to run on the H9000. The "Eventide H9000" package is available in Max's Package Manager for both Windows and macOS.

6.5.1. gen~ Setup

Install and launch the package. The H9000watcher patch will appear.

Click `script npm install`, give it a moment to run, then click `script start`.

You will now be able to select your H9000 from the list. If it's not in the list, make sure that Emote can detect your H9000, then click the H9000watcher's refresh button.

Enter an unoccupied ID between 10101 and 10199. If you select an ID that is already occupied by an algorithm, then that algorithm will be overwritten.

Your computer must be connected to both the H9000, and the internet.

> **Note**
>
> On macOS, if you are connected to the H9000 via a direct ethernet connection, and to the internet via WiFi, then you must ensure that WiFi is above Ethernet in `System Preferences > Network > ... > Set Service Order`

In Max's Package Manager, while viewing the Eventide H9000 package, click `Show in Filebrowser`, then double-click `ringmod.maxpat`, then click the `exportnotifier H9000watcher` message box, then click the `exportcode` message box. The first time you run it, it's going to pop up a dialog that lets you choose where the source code goes. Just accept the default location.

Clicking `exportcode` will send the gen~ patch to Cycling '74's server, then back to your computer and to the selected H9000. While this occurs, there will be status updates in the H9000watcher patch. The algorithm will appear on your H9000 at the ID set in the H9000watcher patch. Open Emote and load the algorithm, or select it from the front panel of your H9000.

6.5.2. Auto-Wah Demo

The "Eventide H9000" package includes a demonstration of a comprehensive auto-wah that uses a trio of gen~ LFOs in Max to modulate gen~ effects on the H9000.

In Max's Package Manager, while viewing the Eventide H9000 package, click `Show in Filebrowser`, right-click `Gen_Auto-wah.maxproj`, select `reveal file`, and double-click the `maxproj` in your OS's file browser. The gen~ auto-wah demo will appear. If you have any questions about this demo, feel free to post in this forum thread, or email support@eventide.com

6.5.3. Sharing gen~ Algorithms

If you would like to share your gen~ algorithm with other H9000 users, you may use the Dump Data feature to export a `.9ka` file that users can upload directly to their H9000 via Emote. To access the Dump Data feature, insert a FAT32-formatted USB drive into the H9000, then navigate to Emote's Devices View, click `UPDATES`, then click `DUMP DATA`. The `.9ka` file will be exported to the USB drive.

6.5.4. Additional gen~ Resources

- gen~ documentation
- gen~ forum
- Example patches: `Max app > menubar > Help > Examples > gen`
- Search on GitHub
- Eventide forum

G. LEGALESE

7.1. Safety and Warranty

Safety Information

- Do **not** remove any covers or panels from the unit when the power is connected.
- No operator access to the internals of the unit is permitted; servicing must be performed by qualified personnel only.
- The unit must **not** be operated with a damaged or ungrounded power cord.
- Suitable ventilation must be provided for the unit at all times. In particular, the side vents must **not** be obstructed. It is best if there is an inch or more clearance between the top of the H9000 and the bottom of the units above and below.

Limited Warranty

The Eventide H9000/H9000R unit covered by this warranty is built to exacting quality standards and should give years of trouble-free service. If you are experiencing problems which are not cleared up in this manual, your recourse is this warranty.

What the Warranty Does and Does Not Cover

Eventide Inc. warrants the above-identified unit to be free from defects in workmanship and material under normal operation and service for a period of one year from the date of purchase, as detailed below.

At our discretion within the warranty period, we may elect to repair or replace the defective unit. This means that if the unit fails under normal operation because of such defect, we will repair the defective unit at no charge for parts or labor. We also assume a limited responsibility for shipping charges, as detailed below.

The warranty does not extend beyond repair or replacement as stated herein and in no event will we be responsible for consequential or incidental damages caused by any defect, and such damages are specifically excluded from this warranty. Our sole obligation is to repair or replace the defective unit as described herein.

The warranty **DOES NOT COVER** any damage to the unit regardless of the cause of that damage. The unit is a complex piece of equipment that does not react well to being dropped, bounced, crushed, soaked or exposed to excessively high temperatures, voltages, electrostatic or electromagnetic fields. If the unit is damaged for these or similar causes, and the unit is deemed to be economically repairable, we will repair it and charge our normal rates.

The warranty **DOES NOT COVER** shipping damage, either to or from Eventide. If you receive a new unit from us in damaged condition, notify the carrier and us; we will arrange to file an insurance claim and either repair or exchange the unit.

If you receive a new unit from a dealer in damaged condition, notify the dealer and the carrier.

If we receive the unit from you with apparent shipping damage, we will notify you and the carrier. In this case, you must arrange to collect on any insurance held by you or your carrier. We will await your instructions as to how to proceed with the unit, but we will charge you for all repairs on damaged units.

Who is Covered Under the Warranty

The warranty applies to the original purchaser of a new unit from Eventide or an Authorized Eventide Dealer. Demo units are also covered by this warranty under slightly different circumstances (see below).

Units that are used, or have been used as part of a rental program, are not covered under any circumstances.

It is your responsibility to prove or to be able to prove that you have purchased the unit under circumstances which activate the warranty. A copy of your purchase invoice is normally necessary and sufficient for this.

If you have any questions about who is an Authorized Eventide Dealer, call us. Units with the serial number plate defaced or removed will not be serviced or covered by this warranty.

When the Warranty Becomes Effective

The one-year warranty period begins on the day the unit is purchased from an Authorized Eventide Dealer or, if the unit is drop-shipped from Eventide, on the day shipped, plus a reasonable allowance for shipping delays. This applies whether or not you return your warranty registration form.

Warranty Information

When we receive a unit, this is how we determine whether it is under warranty:

1. If the unit was shipped from our factory within the past calendar year, we assume that it is under warranty unless there is evidence to the contrary, such as its having been sold as used or rented, etc.

2. If the unit was shipped from our factory more than a calendar year ago, we assume it is not under warranty unless:
- There is a warranty registration form on file showing that it has been purchased within the past year under appropriate conditions.
- You send a copy of your purchase invoice indicating warranty status along with the unit.

3. If the unit was used as a demo, the warranty runs from the date that it was received by the dealer. The original purchaser gets the unexpired portion of that warranty.

When you send a unit for repair, you should indicate whether or not you believe it to be under warranty. If you do not say the unit is under warranty, we will charge you for the repair and we will not refund unless the charge was caused by an error on our part. If you believe the unit to be under warranty and you do say it is but we disagree, you will not incur any charges until the dispute is resolved.

Reading the above, you can see that it is to your advantage to send in the warranty registration form when you purchase the unit. If we know who you are, we can send you updates and notifications, and advise you of our new products. It will also enable you to receive pre-shipment of certain parts.

Who Performs Warranty Work

The only company authorized to perform work under this warranty is Eventide Inc., Little Ferry, New Jersey. While you are free to give personal authorization to anyone else (or to work on it yourself), we will not honor claims for payment for parts or labor from you or from third parties.

However, we and our dealers do try to be helpful in various ways. Our dealers will assist, usually without charge during the warranty period, in:

- Determining whether there is a problem requiring return to the factory.
- Alleviating user error or interconnection problems that may be preventing the unit from operating to its full capability.

We are available for email and telephone consultation if the dealer is unable to assist.

If a part is found to be defective during the warranty period and you wish to replace it yourself, we will normally ship the part immediately at no charge, provided your warranty registration form is on file. We reserve the right to request that the defective part be returned to us.

Shipping Within the 50 United States

You are responsible for getting the unit to our door at no cost to us. We cannot accept collect or COD shipments.

We will return the unit to you prepaid, at our expense, using an expeditious shipping method, normally United Parcel Service. In areas not served by UPS we will ship by US Mail.

If you are in a hurry and want us to use a premium shipping method (such as air express, next day air, etc.), be sure you tell us and agree to pay shipping charges collect. If you specify a method that does not permit collect or COD charges, remit sufficient funds to prepay shipping.

Shipping Outside the 50 United States

If you purchased the unit from a dealer in your country, consult with the dealer before returning the unit.

If you wish to return the unit to us, please note the following:

1. The unit must be prepaid to our door. This means that you are responsible for all shipping charges, including customs brokerage and duties. When a unit is shipped to us it must be cleared through United States Customs by an authorized broker. You must make arrangements for this to be done. Normally, your freight forwarder has a branch in the United States, which can handle this transaction. If you want our assistance, you must notify us before shipping the unit for repair, giving full details of the shipment, and including a minimum of $250.00 in US funds to cover the administrative and brokerage expenses. Any balance will be applied to the repair charges or refunded. If a balance is due to us, we will request a further prepayment.

2. All shipments will be returned to you collect. If this is impossible because of shipping regulations or money is due us, we will request prepayment from you for the appropriate amount.

3. All funds must be in US dollars. Payment may be made by check drawn on any bank in the US, or by telegraphic funds transfer to our bank. If you send US currency, be sure that it is sent by a method you can trace, such as registered mail. If you wish to pay by Letter of Credit, be sure that it affords sufficient time for work to be performed and the L/C negotiated, and that it is free from restrictive conditions and documentation requirements.

4. We reserve the right to substitute freight carriers. Although we will attempt to honor your request for a specific carrier, it is frequently necessary to select a substitute because of difficulties in communication or scheduling.

This warranty gives you specific legal rights and you may also have other rights, which vary from location to location.

7.2. Federal Communications Commission Statement

FC This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference.
2. This device must accept any interference received, including interference that may cause undesired operation.

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates—and can radiate—radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a installation.

If the equipment does not cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and the receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Warning

This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures to correct this interference.

Class A ITE

Class A ITE is a category of all other ITE which satisfies the class A ITE limits but not the class B ITE limits. Such equipment should not be restricted in its sale but the above warning shall be included in the instructions for use.

CE Class A (EMC)

CE This product is herewith confirmed to comply with the requirements set out in the Council Directives on the Approximation of the laws of the Member States relating to Electromagnetic Compatibility Directive 2004/108/EEC.