

# Getting Started with



# Copyright and Trademarks

**MODE** and this manual are © 2004 Cycling '74. **MODE** uses Max/MSP, published by Cycling '74. Max is copyright © 1990-2004 Cycling '74 and IRCAM, l'Institut de Recherche et Coordination Acoustique/Musique. MSP is copyright © 1997-2004 Cycling '74—All rights reserved. Portions of MSP are based on Pd by Miller Puckette, © 1997-2004 The Regents of the University of California. MSP and Pd are based on ideas in FTS, an advanced DSP platform © IRCAM.

Pluggo and MSP are trademarks of Cycling '74. VST and Cubase are trademarks of Steinberg Soft- und Hardware GmbH. Digital Performer, Performer, and Mark of the Unicorn are trademarks of Mark of the Unicorn, Inc. Digidesign and Pro Tools are trademarks of Digidesign, a division of Avid Technology, Inc. Ableton and Live are trademarks of Ableton AG.

# Introduction

## About This Manual

Welcome to **MODE**, the premium plug-in package by Cycling '74 and CreativeSynth. This *Getting Started* manual covers installation of the program, operation of the interface features common to all Pluggo-based plug-ins, synchronization and modulators.

Do you need to read this entire manual? We definitely recommend reading through the description of the installation, authorization and registration processes. The rest of this manual may cover some material with which you are already familiar. However, it's pretty short and discusses a few things that might not be obvious to the new user. For instance, did you know you can undo changes to parameters in the MODE plug-ins? Look in the *Plug-in User Interface* chapter for the details.

The documentation assumes that you are familiar with the basic operations of the program you'll be using to host the MODE plug-ins. We do mention a few things that are specific to using pluggo with applications such as Cubase, Pro Tools, Digital Performer, Logic Audio and Live, but very little is said about creating or importing audio tracks or using audio I/O.

***The Book of MODE***, which describes each of the included plug-ins, is in electronic format. It's found in the *Pluggo Applications/Documentation/* folder on the Applications folder on the hard disk. Plug-ins generally come with example preset effect programs (when it makes sense) that can be used to determine some of the more esoteric uses of the effect or instrument.

## Basic Features of MODE

- A lot of plug-ins (23 of them)
- Support for host synchronization in VST and Audio Units applications.
- Modulator plug-ins that control the parameters of other plug-ins
- Instrument plug-ins that respond to sequenced and live MIDI input

## System Requirements

MODE requires:

- A Macintosh equipped with a G4 processor running at 800 Mhz or faster is recommended.
- Mac OS X 10.2 or later
- A sequencer application that hosts VST, RTAS, or Audio Units plug-ins.
- At least 256 MB of memory, although the exact amount depends on the system version and the host sequencer you're using.
- At least 25MB of hard disk space

## **Where to Get Help and Support**

If you encounter problems when using MODE or the Pluggo technology, the following resources are available:

- CreativeSynth.com, the MODE developers, are on-line at <http://www.creativesynth.com>
- The Pluggo FAQ document, on-line at <http://www.cycling74.com/support/questionspl.html>
- The Pluggo troubleshooting guide, on-line at <http://www.cycling74.com/products/pluggo/troubleshooting.html>
- e-mail to [support@cycling74.com](mailto:support@cycling74.com)
- You may telephone our customer support line at (415) 869-2812
- If you have any questions related to installation and authorization of the software, you may telephone our office at (415) 974-1818, generally open from 10AM to 4PM Pacific time.

E-mail or telephone technical support is only available to customers who have purchased pluggo. If you did not purchase your copy of pluggo directly from Cycling '74, please send in the registration card in your package or register at [www.cycling74.com/register](http://www.cycling74.com/register).

## **A Little Background Information on MODE and Pluggo**

The MODE plug-ins use the Pluggo technology created by Cycling '74. A bit of background regarding Pluggo might be in order.

Pluggo is a runtime shell that uses the Max object-oriented graphical programming environment from Cycling '74. pluggo's signal processing capabilities are provided by the MSP audio extensions to Max. Using Max and MSP, you can write your own audio and MIDI plug-ins.

The authors of Max and MSP are Miller Puckette and David Zicarelli. The MODE plug-ins and supporting externals were developed by Darwin Grosse and Chris Randall. Additional programming was provided by Gregory Taylor and Adam Schabtach.

# Installing MODE

Installing pluggo involves two steps. First, running the MODE Installer from the CD-ROM or your software download folder, and second, running the Authorize Pluggo application to authorize the software.

## What is Going to be Installed?

The MODE Installer will put the plug-ins in the /Library/Audio/Plug-ins/VST/Pluggo folder. It also places support files in /Library/Application Support/Cycling '74 and /Library/CFMSupport. None of these folders should be moved or renamed.

Audio Unit support is added to the Library/Audio/Plug-Ins/Components folder.

If you're installing the RTAS version of pluggo, a "stub" plug-in called Pluggo-RTAS is installed in the Library/Application Support/Digidesign/Plug-ins folder.

The Authorize MODE and Pluggo AU Scan applications are installed into the Pluggo Applications folder inside the Applications folder. In addition, documentation is placed in the Pluggo Applications folder.

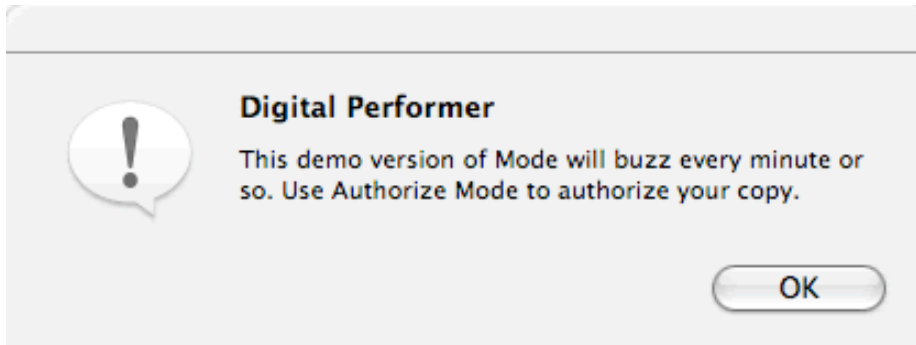
## The MODE Installer

When you run the MODE Installer, you'll first see a license agreement covering the software and its documentation. Please read the agreement carefully and click Agree if you are willing to abide by its terms.

If you click Install, the plug-ins and support files will be installed in the the appropriate locations. After the installation is complete, an optional "AU SCAN" application is provided, which registers all of the MODE plug-ins for AudioUnit support. If you use any hosts that support AudioUnit plug-ins (Digital Performer, Logic, GarageBand), you will want to allow this program to run.

## Authorizing MODE

When you install MODE, it's in demo mode. You'll see the following dialog box when you open the first MODE plug-in within your sequencer:



In order to gain unlimited use of MODE, you need to run the Authorize MODE application and enter a valid authorization code.

If you purchased the packaged version of MODE, your serial number, authorization code and Registration ID are inside the pluggo box.

Please keep your authorization code, serial number, and Registration ID in a safe place.

If you downloaded MODE, you can purchase an authorization code from Cycling '74. The most convenient way to do so is by using a credit card on our secure on-line order form. We will send you the authorization code after receiving your order. Here's the procedure:

- Run Authorize MODE, found in your Pluggo Applications folder in the Applications folder of the drive where you installed MODE. The opening screen will ask you whether you want to purchase a MODE authorization code, or use an authorization code you have received to authorize MODE. When you select the option to purchase an authorization code and click Next, you will see a screen with a Registration ID.
- If you click on the option to purchase an authorization code and click the Web Page button or press the return key, the Authorize MODE program will start up your web browser and automatically take you to the online MODE purchase form. The program will also automatically enter your Registration ID on the online purchase form. Enter

the additional relevant information on the form and click to purchase an authorization code.

- Once we process your order, you'll receive e-mail with an authorization code. Copy the authorization code text in the e-mail message to the clipboard so you can paste it into the authorization code page.

If you prefer to purchase a packaged version of MODE, visit your local retailer or use the order form on the Cycling '74 web site to buy one directly from us. No Registration ID is needed to purchase the packaged version—the ID is found inside the box, and you'll need to reinstall MODE from the CD in the packaged version in order to authorize the software.

## Entering Your Code

- Run Authorize MODE. Select the option to enter an authorization code and click Next. You will see a dialog box which shows your Registration ID and provides a space for entering your authorization code:



Authorize Mode

Your registration ID is:  
XXXX-XXXX-XXXX-XXXX

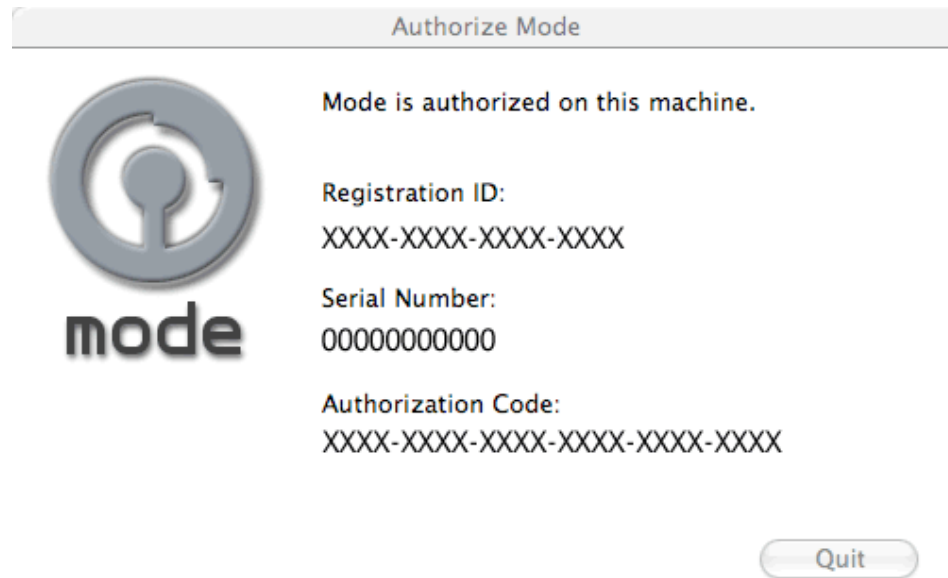
Please enter your authorization code:

Paste Authorization Code

Previous Next

If you see an error message instead, you may need to re-enter your authorization code, or re-install the software before you can authorize.

- Click Next after entering your authorization code. You should then a dialog box which contains your Registration ID, your Customer Number, and your Authorization code. The pluggo authorization serves to authorize every host application that uses the same system on your computer.



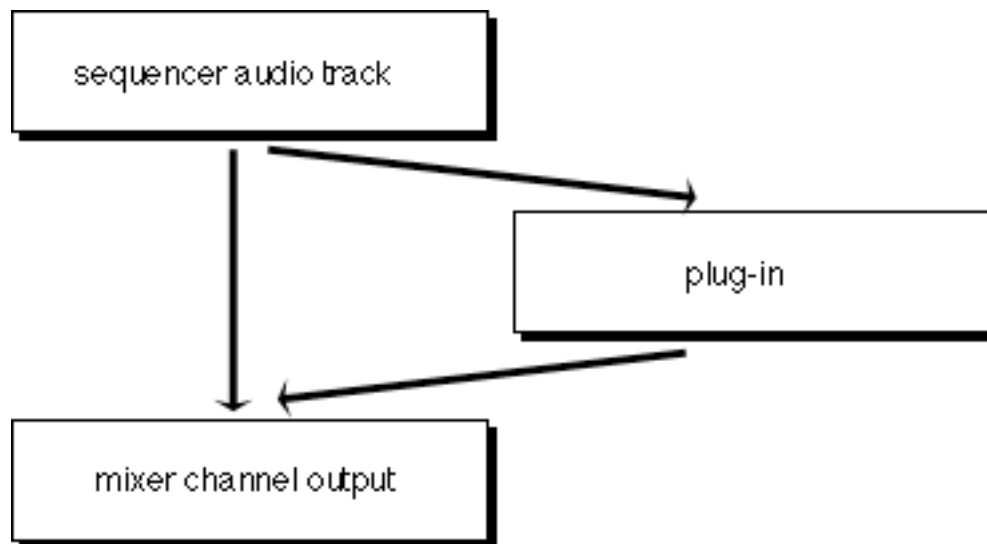
If you run Authorize MODE again and your computer is already authorized, you'll see the above dialog.

# Using MODE

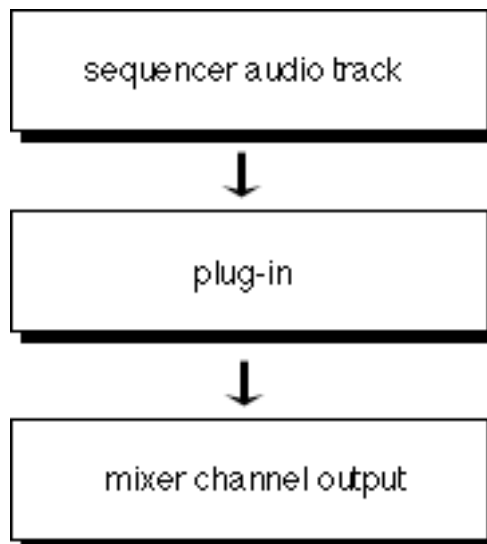
## Inserting Plug-ins Within Your Sequencer

In this chapter, we'll cover the steps for inserting MODE plug-ins with Cubase, Digital Performer, Live, Logic Audio, Max/MSP and Pro Tools.

When we say "inserting" a plug-in, we mean placing it into the audio processing path of a mixer. There are two kinds of insertions. In one kind of insertion, the plug-in adds its signal to the input as a kind of bus. Cubase Send Effects are an example of this kind of insertion.



In another kind of insertion, used in Pro Tools, Logic Audio, Digital Performer, and in the Master and Insert Effects of Cubase, the plug-in replaces its input with its output.



After installing MODE you will find an assortment of new plug-ins available to your host.

The MODE plug-ins can be opened in either a mono, stereo, or multi-channel context. However, a mono plug-in may not do what you want if you try to process a stereo signal—you may just hear processing on the left channel.

It's also possible in some sequencers to insert plug-ins into a context where they accept a mono input signal and produce a stereo or multi-channel output signal. Typically, to insert a stereo plug-in, the channel in the mixer you're using for the plug-in has to be a stereo channel.

## Using Virtual Instruments

The MODE collection includes a selection of virtual instruments—plug-ins that receive MIDI and generate only audio output.

These virtual instruments are accessible to host sequencer applications in different ways which vary according to the specific host application. In Cubase, these plug-ins are called VST instruments, and have their own Instrument panel. In Logic Audio, virtual instruments are choices from a set of available Audio Instruments. Digital Performer uses instrument tracks to host virtual instruments. In Pro Tools, virtual instruments appear on the plug-in menus along with audio plug-ins.

Typically, you'll need to create a new track for your virtual instrument, and then route MIDI data from your synthesizer or a pre-recorded MIDI track or file to the new virtual instrument. For more specific details, see the documentation for your specific host application.

## Using Virtual Instruments in Max/MSP

You can insert and use MODE virtual instrument plug-ins in Cycling '74's Max/MSP environment in the same way that you use audio plug-ins. The MSP `vst~` object loads pluggo virtual instrument plug-ins in the same way as any other VST plug-in, and virtual instrument parameters can be changed in Max/MSP via messages to the `vst~` object.

## Plug-in Automation

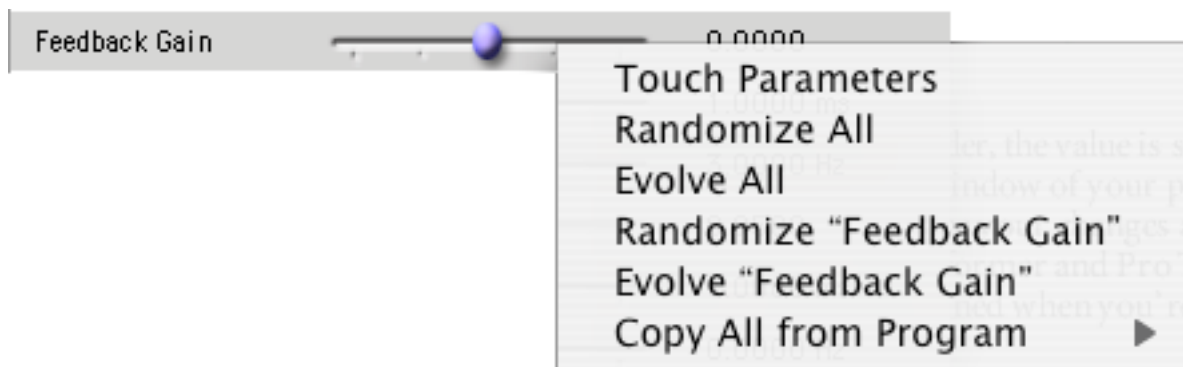
Some sequencers allow you to capture changes you make to MODE plug-ins while the music is playing back, then replay these changes in time with the music. The

most common application of this feature is automated mixdown—meaning you essentially get more hands to move mixing board controls while the music plays back.

These sequencers let you automate effects parameter changes either by recording fader changes in real time or by inserting and graphically editing control envelopes for pluggo effects parameters. For more information on automation, refer to the manual of your specific host application.

## The Parameter Change Pop-up Menu

Hold down the command key and click on a control. You'll get a pop-up menu that looks something like this:



The Parameter Change pop-up menu contains commands that deal with parameter values in the plug-in.

*Touch Parameters* is used to send out messages to the host sequencer that describe the current position of all the controls for a plug-in. It is used with Digital Performer, Pro Tools and Logic Audio to aid in plug-in automation. It tells the host that all parameters have been “changed” to their current value, so that host can record these changes. This command has no effect on the actual values of any plug-in parameters.

*Undo Last Change* allows you to go back to the previous setting of a parameter that you changed with a slider. It can also return you to the state the plug-in was in before selecting a new program.

Choose *Undo Last Change* again to return to the change before you undid it. (There is only one level of undo available.)

It doesn't matter what slider you command-click on to activate the pop-up menu if you want to choose *Undo Last Change*. Only the most recent change to any slider is undoable, so the slider that was changed last will be reset.

*Randomize All* will set the parameters to random values. This can be useful in finding effect settings you never would have been able to come up with if you had adjusted the sliders one at a time. Not all parameters may be affected however; the developer of the plug-in can keep certain parameters (such as output gain) from being randomized that would simply be irritating if they were changed.

*Evolve All* will nudge each parameter by a random amount up to five percent greater or less than its current value.

If you click on a particular slider, you'll get two additional menu items specific to the parameter controlled by the slider. Randomize "Parameter Name" will set the value of the parameter whose slider you clicked on to a random value. Evolve "Parameter Name" will nudge the parameter whose slider you clicked on by a random amount up to five percent greater or less than its current value.

*Copy All from Program* lists the names of all of the programs for this plug-in. When you choose a program name from this menu, you copy the settings from the selected program into the current program. You can Undo this action if desired.

## Using the View Menu

At the top of the MODE edit window, you'll see a menu labeled View. Click on this menu to see other pages or views of the plug-in's interface or to access a plug-in's presets.

The Controls view is available with all MODE plug-ins, and contains the user interface for programming the synthesizers and effects.

- Choose *Pluggo Info* from the View menu. This is the Pluggo subsystem's About Box. Here's a secret: clicking on the pluggo character is the same as Randomize All from the Parameter Change pop-up menu. The Pluggo Info view will appear and behave slightly differently depending on the size of the plug-in.

- Some plug-ins provide a *Messages* view. This window may provide diagnostic information that could be helpful if the plug-in does not appear to be working. For instance, you might see a message something like this after opening a plug-in:

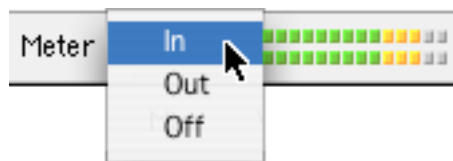
error: plugin~: no such object

This particular error indicates an incorrect or corrupted MODE installation.

Note that the contents of the Messages view is common to all pluggo-based plug-ins. In other words, if you've already loaded a plug-in that reported information or errors in the Messages view, you'll see the messages it generated in the Messages view of any subsequent plug-in you insert.

## The Level Meter

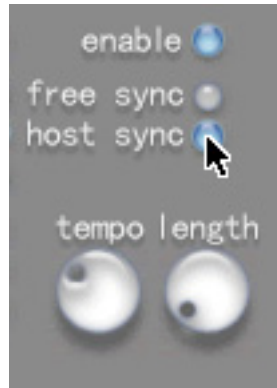
The MODE plug-ins contains a handy level meter that tells you whether the plug-in is getting any input signal level or producing an output signal. Normally the meter is set to display the level of the input signal, as shown below:



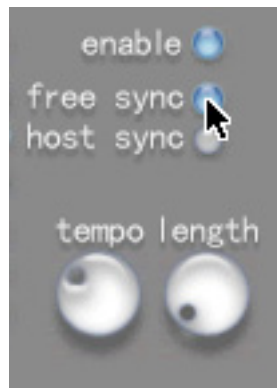
- Choose Out to have the meter display the plug-in's output signal level.
- Choose Off to turn the meter off. Some plug-ins may turn the meter off by default. As with any on-screen level meter, the pluggo meter does consume a tiny amount of the CPU available for signal processing.

# Synchronizing Plug-ins

Synchronization of time-based plug-ins (MODE.MONO, MODE.POLY and the MODE.SPIN plug-ins, to be precise) are based on the availability of plug-in host synchronization. If the host provides sync, selecting “Host Sync” within the tempo control sections will allow your tempo to “lock” to the host’s timing.



If your host does not provide host synchronization, or you want to use a free-running tempo, select “Free” as the sync option (or, in some cases, deselect the Host Sync option) and freely set a tempo for the plug-in.

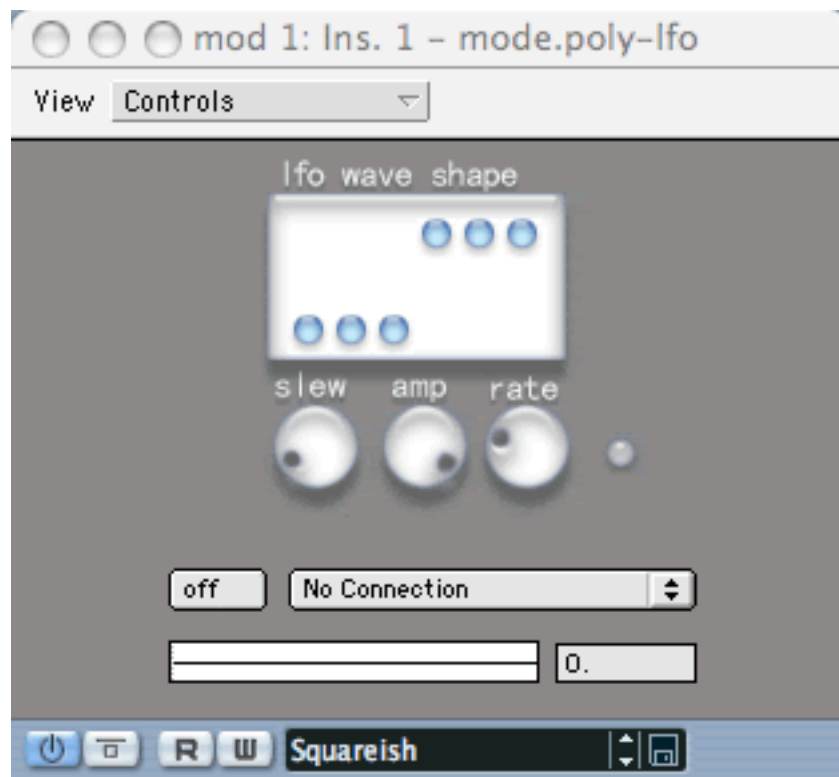


# Using Modulator Plug-ins

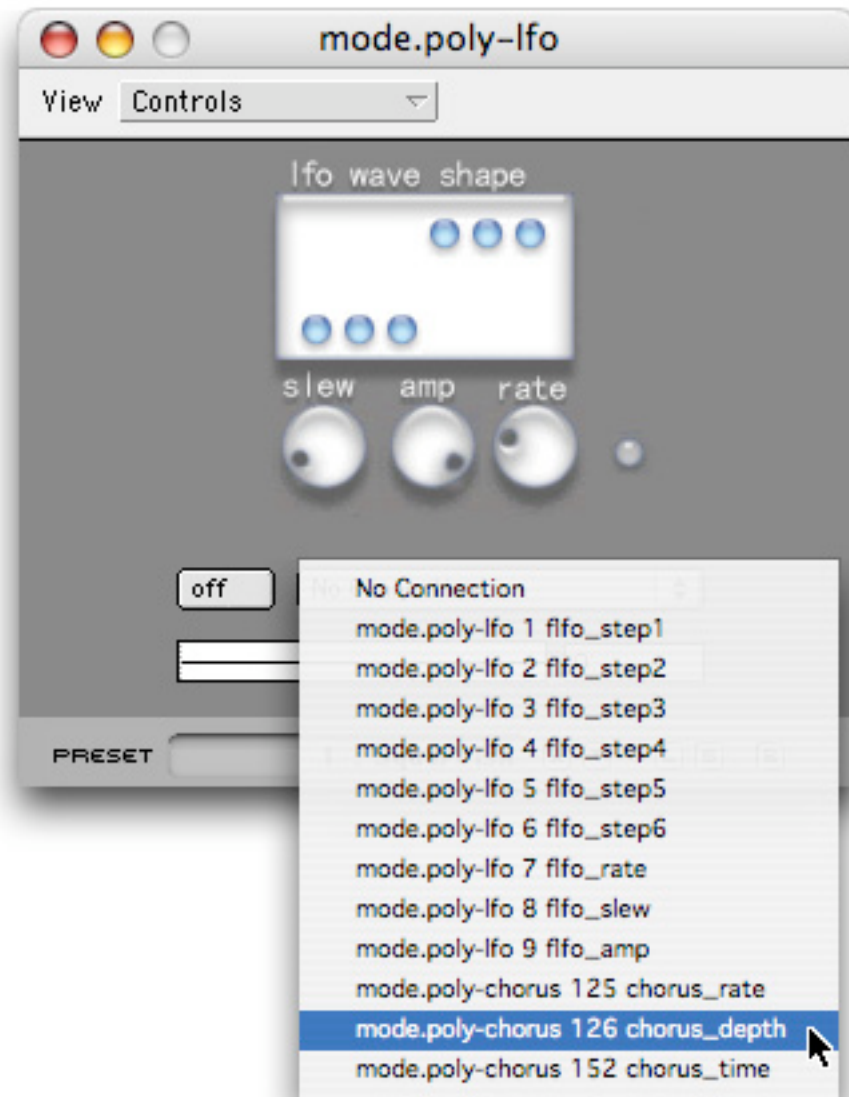
MODE's Pluggo-based Modulators are plug-ins that do not process any audio. Instead they are designed to modify the parameters of other plug-ins. The MODE collection, like Pluggo, provides several different Modulators. We're going to look at one of the simpler ones called *MODE.POLY-LFO*. It lets you set up an LFO modulation that can be used to control other plug-in parameters.

Modulator plug-ins pass audio signals through them; indeed, most ignore their audio inputs completely. So you can insert them anywhere without affecting the signal path of your mixer.

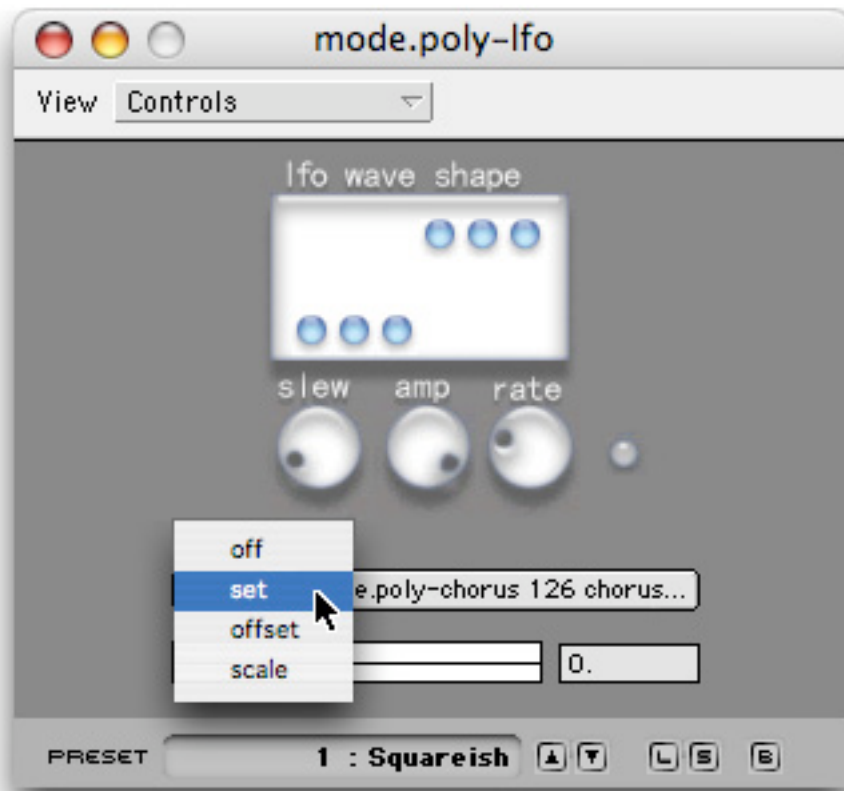
- Set a channel in your sequencer's mixer to handle an audio signal and insert both the *MODE.POLY-LFO* and *MODE.POLY-CHORUS* plug-ins on that channel. You can put *POLY-LFO* before or after *POLY-CHORUS* in the effect chain, it doesn't matter.
- Open the *MODE.POLY-LFO* edit window.



- Click on the top-right menu, and choose mode.poly-chorus's depth setting.



Then choose Set from the menu immediately to its left. This assigns the output of the LFO to modify *poly-chorus's* depth setting.



- Open the *POLY-CHORUS* plug-in's edit window. You should see the Depth parameters moving around in step with the LFO, and (if you are playing the track) you should also hear a change in the audible result. Experiment with different Output Ranges in the *POLY-LFO* window (just click to drag out a range); this will affect the ratio of LFO value to parameter change. For the depth settings, a value that's too high will not sound like chorusing, so it's best to limit the Output Range to the middle of the total range, like this:

