Phosphor User's Guide
Audio Damage, Inc.
Release 3.0

15 May 2020

Credits

Software Design and Construction, Documentation
- Chris Randall
- Adam Schabtach

Field Testing, Version 3.0
- Eric Beam
- Don Gunn
- Steve Hamman
- Christian Puffer
- Joshua Schnable
- Simon-Claudius Wystrach

Field Testing, Version 2.0
- Eric Beam
- Steve Hamman
- Jeff Laity
- Ken Flux Pierce
- Christian Puffer
- Joshua Schnable
- Simon-Claudius Wystrach

Field Testing, Version 1.0
- Jens Ågren
- Steve Hamann
- Jeff Laity
- Mitchell Sigman

Made Possible By
- Elle
- Tracie

License Agreement

BY INSTALLING THE SOFTWARE, YOU ARE CONSENTING TO BE BOUND BY THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, THEN RETURN THE PRODUCT TO THE PLACE OF PURCHASE FOR A FULL REFUND.

Single User License Grant: Audio Damage, Inc. ("Audio Damage") and its suppliers grant to Customer ("Customer") a nonexclusive and nontransferable license to use the Audio Damage software ("Software") in object code form solely on a single central processing unit owned or leased by Customer.

Customer may make one (1) archival copy of the Software provided Customer affixes to such copy all copyright, confidentiality, and proprietary notices that appear on the original.

EXCEPT AS EXPRESSLY AUTHORIZED ABOVE, CUSTOMER SHALL NOT: COPY, IN WHOLE OR IN PART, SOFTWARE OR DOCUMENTATION; MODIFY THE SOFTWARE; REVERSE COMPILE OR REVERSE ASSEMBLE ALL OR ANY PORTION OF THE SOFTWARE; OR RENT, LEASE, DISTRIBUTE, SELL, OR CREATE DERIVATIVE WORKS OF THE SOFTWARE.

Customer agrees that aspects of the licensed materials, including the specific design and structure of individual programs, constitute trade secrets and/or copyrighted material of Audio Damage. Customer agrees not to disclose, provide, or otherwise make available such trade secrets or copyrighted material in any form to any third party without the prior written consent of Audio Damage. Customer agrees to implement reasonable security measures to protect such trade secrets and copyrighted material. Title to Software and documentation shall remain solely with Audio Damage.

LIMITED WARRANTY. Audio Damage warrants that for a period of ninety (90) days from the date of shipment from Audio Damage: (i) the media on which the Software is furnished will be free of defects in materials and workmanship under normal use; and (ii) the Software substantially conforms to its published specifications. Except for the foregoing, the Software is provided AS IS. This limited warranty extends only to Customer as the original licensee. Customer's exclusive remedy and the entire liability of Audio Damage and its suppliers under this limited warranty will be, at Audio Damage or its service center's option, repair, replacement, or refund of the Software if reported (or, upon request, returned) to the party supplying the Software to Customer. In no event does Audio Damage warrant that the Software is error free or that Customer will be able to operate the Software without problems or interruptions.

This warranty does not apply if the software (a) has been altered, except by Audio Damage, (b) has not been installed, operated, repaired, or maintained in accordance with instructions supplied by Audio Damage, (c) has been subjected to abnormal physical or electrical stress, misuse, negligence, or accident, or (d) is used in ultrahazardous activities.

DISCLAIMER. EXCEPT AS SPECIFIED IN THIS WARRANTY, ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS, AND WARRANTIES INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE, ARE HEREBY EXCLUDED TO THE EXTENT ALLOWED BY APPLICABLE LAW.

IN NO EVENT WILL AUDIO DAMAGE OR ITS SUPPLIERS BE LIABLE FOR ANY LOST REVENUE, PROFIT, OR DATA, OR FOR SPECIAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL, OR PUNITIVE DAMAGES HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE EVEN IF AUDIO DAMAGE OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. In no event shall Audio Damage's or its suppliers' liability to Customer, whether in contract, tort (including negligence), or otherwise, exceed the price paid by Customer. The foregoing limitations shall apply even if the above-stated warranty fails of its essential purpose. SOME STATES DO NOT ALLOW LIMITATION OR EXCLUSION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES.

The above warranty DOES NOT apply to any beta software, any software made available for testing or demonstration purposes, any temporary software modules or any software for which Audio Damage does not receive a license fee. All such software products are provided AS IS without any warranty whatsoever.

This License is effective until terminated. Customer may terminate this License at any time by destroying all copies of Software including any documentation. This License will terminate immediately without notice from Audio Damage if Customer fails to comply with any provision of this License. Upon termination, Customer must destroy all copies of Software.

Software, including technical data, is subject to U.S. export control laws, including the U.S. Export Administration Act and its associated regulations, and may be subject to export or import regulations in other countries. Customer agrees to comply strictly with all such regulations and acknowledges that it has the responsibility to obtain licenses to export, re-export, or import Software.

This License shall be governed by and construed in accordance with the laws of the State of Colorado, United States of America, as if performed wholly within the state and without giving effect to the principles of conflict of law. If any portion hereof is found to be void or unenforceable, the remaining provisions of this License shall remain in full force and effect. This License constitutes the entire License between the parties with respect to the use of the Software.

Introduction

Phosphor is a polyphonic software synthesizer plugin modeled on the alphaSyntauri hardware synthesizer. Introduced in 1979, the alphaSyntauri used an Apple][microcomputer as its central processor and user interface. While quite modest by contemporary standards, the alphaSyntauri can create a surprising wealth of sounds easily, thanks to its simple but flexible approach to synthesis.

Rather than mimicking the small number of wave shapes available from the oscillators found in analog synthesizers, the alphaSyntauri's oscillators used tables of numbers—essentially samples of a single cycle of an audio wave. The tables are filled using a process known as *additive synthesis*. The basic idea of additive synthesis is that any sound wave can be created by adding together many sine waves of different frequencies. Each sine wave has a pure sound with no harmonics, but if you add a bunch of them together you can create complex sounds. If you use enough sine waves, and if you control their relative loudness precisely over the duration of a note, you can recreate just about any sound.[1]

Since the alphaSyntauri was limited by the computational power available in its day, it took a relatively simple approach to additive synthesis. Each oscillator's wave shape was created by adding together 16 sine waves, which we'll henceforth refer to as *partials*. The wave shape could not change over the duration of a note, but each oscillator's overall loudness was controlled by a separate envelope generator, and hence the timbre of the note could be varied by mixing the two oscillators dynamically. Phosphor reproduces this system faithfully and adds several tricks of its own.

Phosphor harkens back to a time when high-quality digital synthesis was available only to people with huge budgets or access to academic research facilities, but low-quality digital synthesis was finding its way into the public ear through quarter-munching video games and 8-bit home computers. We hope you enjoy using it as much as we enjoyed building it.

New in Version 3.0

Version 3 of Phosphor adds several new features and refinements:

- Much more modulation routing
- Independent LFOs per note, with graphical wave-shape displays
- Transposition and fine-tuning controls for each oscillator, replacing the fixed-offset control
- Support for MIDI Polyphonic Expression (MPE) controllers
- Multi-pane, resizable user interface
- Support for non-equal-tempered intonations via TUN files
- Presets updated to take advantage of new MIDI features
- A few other little improvements here and there

Phosphor 3 has a different name and internal identifier than versions 1 and 2. This means that you can install Phosphor 3 alongside whatever version you are currently using and continue to use the older version in your existing projects. Version 3 cannot be directly substituted for previous versions in existing projects and cannot read preset files created by version 1. However, it can read preset files created with version 2.

Was New in Version 2.0

If you happen to be updating from the first version of Phosphor, here is what changed with Version 2.0:

- More partial sliders, for a choice of 16, 32, or 64 per oscillator
- Commands for initializing the oscillators with common synthesizer waveforms
- A selection of presets from the previous version, some all-new presets to take advantage of the new features, and a selection of Designer presets from Ken Flux Pierce
- A new user interface, appropriate to contemporary display technologies
- VST 3 and AAX compatibility
- A cross-platform XML-based preset file system
- Built with up-to-date code libraries for better host compatibility and future-proofing

[1] For an in-depth explanation of how this works, see http://en.wikipedia.org/wiki/Additive_synthesis

System Requirements

The following table summarizes the operating system requirements and provided formats for Phosphor:

Operating System	Minimum Version	Formats
macOS	10.11	AudioUnit, VST2, VST3 and AAX
Windows	8.1 x64	VST2, VST3 and AAX
iOS	iOS 11 or iPadOS 13	AUv3, standalone app with IAA

To use Phosphor, you'll need a host application such as Ableton Live, Steinberg Cubase, Apple Logic, Avid ProTools, etc[2]. We assume that you are familiar with using plugins with your host. If you have general questions about using plugins with your host, please refer to its documentation.

The iOS versions of Phosphor require an iPad; newer models will provide better performance.

Demonstration Version

We encourage you to download and try the demonstration version of Phosphor before purchasing it. The demo version of Phosphor is the same as the regular version, but has the following limitations:

- Presets cannot be saved, nor can parameter values or other settings. This includes the information usually stored by your host DAW. If you save a DAW session with an instance of the demo version of Phosphor, Phosphor will revert to its default state when you reload the session.
- Phosphor will cease to produce audio at all 20 minutes after you add it to your DAW session. You can remove it and add it again, but it will revert to its default state.

If You Don't Like Reading Manuals

You will find a video walkthrough which covers all of Phosphor's features here:

https://youtu.be/j8GuiMoP4mI

[2] Product names and plugin format names are copyrighted by their respective owners.

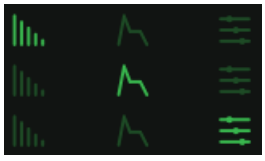
Overview

Phosphor generates sound with a pair of identical *voices*, labeled Voice One and Voice Two. Each voice has a wavetable oscillator and a noise generator, mixed together. Each voice also has an ADSR envelope generator which controls the loudness of that mixed signal. For modulation, Phosphor has a pair of identical LFOs. Each LFO can be routed to either voice, or both. Finally, each voice has a delay line, with filter.

The following screen shot shows Phosphor's user interface. Phosphor's window is resizable; you can adjust it as needed to fit your screen. We will describe Phosphor's controls and operation in the following text. Phosphor is, by design, a relatively uncomplicated synthesizer, so feel free to experiment and return to this manual if you need explanation of any particular feature.

Voice Panels

Phosphor's window displays the controls for each voice in moveable panels. Each voice has three panels, selected by clicking the buttons near the top of the window. The buttons, from left to right, invoke the panels for the wavetable oscillators, ADSRs, and other voice controls.

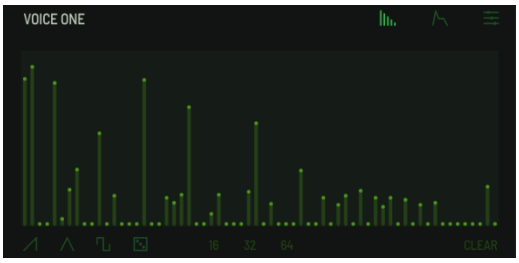


Wavetable Oscillators

Phosphor's core is a pair of audio oscillators. Each oscillator has its own wave table and hence its own timbre. The controls for the wave tables are in the first of the three moveable panels.

The groups of vertical sliders are called the *partial sliders*. The partial sliders control the level of the sine waves used to calculate Phosphor's wave tables, and hence control the tone of the oscillators. Each oscillator has its own set of partial sliders which set the relative amplitudes of each of the partials. The left-most slider sets the level of the *fundamental*, that is, the lowest-frequency partial, which has the frequency of the note you're playing. The next slider

controls a partial whose frequency is twice that of the fundamental (i.e., one octave above the root pitch). The next slider's partial has a frequency of three times that of the fundamental, and so on.



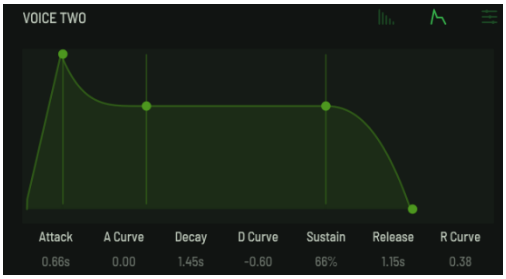
Each oscillator can have 16, 32, or 64 partial sliders. The alphaSyntauri had only 16 partials; hence version 1 of Phosphor also had 16 partials. Adding more partials adds more higher-frequency components to the sound.

The buttons below the partial sliders provide several editing shortcuts. The three buttons at the left set the partials such that the wavetable is filled with a standard synthesizer waveform—sawtooth, triangle, or square. Clicking the die button sets one-third of the partials, chosen at random, to random levels.

The buttons labeled 16, 32, and 64 determine how many partials are used to compute the wavetable. Clicking one of these buttons changes the number of partial sliders correspondingly. Adding more partials increases the brightness and complexity of the oscillator's tone but can also add aliasing noise when the frequency of the partials exceeds half the sampling rate. Aliasing, however, is not always a bad thing.

Envelope Generators

Phosphor employs a pair of envelope generators to shape the volume or loudness of each oscillator over the duration of a note. Since Phosphor's oscillators can create entirely different timbres, using the two envelope generators to vary the relative loudness of each oscillator is one of the keys to producing interesting sound with Phosphor.



Phosphor's envelope generators use the traditional "ADSR" design found in numerous hardware and software synthesizers. Unlike most ADSR envelope generators, Phosphor's EGs let you control the curvature of the Attack, Decay, and Release segments of the envelope.

The second of the three moveable panels holds graphical controls for Phosphor's envelope generators. Click and drag the circular green handles to change the envelope parameters. Drag the handles horizontally to change the times; drag the two middle handles vertically to change the sustain level. You can also drag vertically anywhere near the sustain segment to change its level. (The sustain level is the level at which the envelope remains while you hold down a key on your keyboard.)

If you drag vertically between the handles, or to the left of the first handle, you change the curvature of the envelope segment. The alphaSyntauri's envelopes had only linear shape, but Phosphor's adjustable curvature provides a greater variety of envelope shapes.

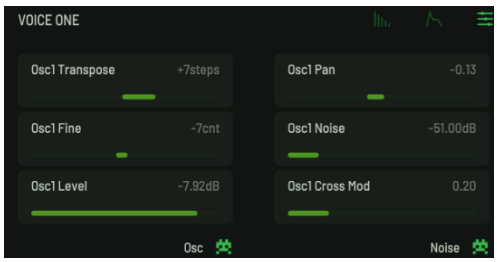
Alternatively, you can click or tap on the numbers below the graphical display to change the envelope settings.

Voice Control Sliders

The third of the three panels house groups of sliders which control several parameters for the voices.

Transpose

The **Transpose** slider changes the pitch of the oscillator, in semitones relative to the pitch set by incoming MIDI notes. You can transpose each oscillator by up to two octaves, either up or down.



Fine

The Fine slider adjusts the oscillator's pitch by small amounts, up to a semitone either sharp or flat. Its value is marked in cents, or 1/100th of a semitone.

Pan

The Pan slider controls the stereo placement of the voice. Moving the Pan slider left and right moves the voice's output in the corresponding direction in Phosphor's stereo output.

Noise

The Noise slider controls the loudness of noise generators mixed with the signals from the oscillators. The noise generators create white noise—noise with a flat frequency spectrum—and are useful for adding bite to percussive sounds or a bit of grunge to the pure sine waves emanating from the oscillators.

Level

The Level sliders control the loudness of the oscillators and the noise sources. Moving the slider to the right makes the oscillator's tone and the noise louder, moving them to the left makes them quieter. Note that you won't hear the noise source unless you move the Noise slider at least somewhat to the right also. If you want just the noise without the oscillator, set all the voice's partial sliders to zero.

Cross Mod

The Cross Mod sliders enable the two oscillators to modulate each other's frequency, creating a much wider range of tones than they create individually. While Yamaha made frequency-modulation (FM) synthesis famous with the DX7, Phosphor draws inspiration from the NED Synclavier, which combined wavetable and FM synthesis.

Moving voice one's Cross Mod slider to the right increases the amount that voice two's oscillator modulates the first oscillator's frequency. This changes the tone produced by the first voice's oscillator (and not the second voice's oscillator). In the usual parlance, voice one's oscillator is the *carrier* and the voice two's oscillator is the *modulator*. As you move the slider to the right, the tone of voice one's oscillator will become brighter and more complex.

It works the other way around for voice two's Cross Mod slider: moving it to the right increases the amount that the voice one's oscillator modulates the second voice's oscillator, and in this instance, we would say that the second voice's oscillator is the carrier and the voice one's oscillator is the modulator. Since both oscillators can modulate each other, we call Phosphor's FM scheme cross-modulation.

The signals used for cross-modulation originate after the envelope generator but before the Level slider. This means two things: first, since the timbre created by frequency modulation changes with the amplitude of the modulating signal, you can use the envelope generator of the modulator to make the tone of the carrier vary over the duration of the note. Second, you can use the level slider of the modulator to reduce or remove its signal from Phosphor's output so that you hear only the changing tone of the carrier.

Vintage Switches

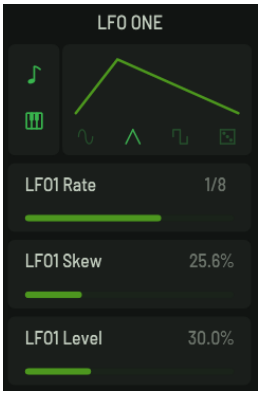
Some of the charm of older digital synthesizers lies in their less than perfect sound-generation hardware. Compared to contemporary synthesizers, instruments of yesteryear had lower sampling rates, lower bit depths, and lower-quality digital-to-analog converters. The resulting noise and distortion weren't considered desirable at the time, but in nostalgic retrospect we've developed a fondness for these quirks.

If you click on the Noise switch, the noise generator reproduces the sound of noise sources found in computer sound generators that were contemporaries of the alphaSyntauri. In this mode, the noise generators have some apparent pitch and will track the keyboard as you play. If you've played old video games, you'll recognize the timbre of the noise generator, particularly near the low end of the keyboard.

If you turn on the Osc switch, the quality of Phosphor's oscillators is reduced to resemble the alphaSyntauri's hardware. The vintage switch introduces aliasing and digital grunge which can add some interesting color to Phosphor's tone.

Low-Frequency Oscillators (LFOs)

Phosphor has two low-frequency oscillators (LFOs) that can be used to modulate parameters of the voices, filters, and delays.



The Rate sliders determine how fast the output of the LFO varies over time. They operate either in units of frequency (Hertz, or cycles per second), or in metrical units. Click the musical-note buttons to switch between the two modes. If sync is off, the LFO's rate can be set from one cycle every 100 seconds (or 0.01 cycles per second, abbreviated 0.01 Hz) to 14 cycles every second (14 Hz).

When sync is turned on, the rate of the LFO is synchronized to the tempo provided by your host DAW. In this mode the Rate slider sets the period of the LFO in metrical units. The range of values is 1/32nd to 1/1 (a whole measure), with dotted and triplet times available. Triplet values are denoted with a "T" after the beat fraction, and dotted values are denoted with a "D".

For example, "1/8 D" indicates a cycle length with a dotted eighth note feel. Phosphor will track tempo changes, saving you from having to adjust its LFOs by hand when you change the tempo of your song.

The four buttons which look like small waveforms work together with the **Skew** slider to control how the LFO's output varies over time. The drawing above the buttons shows the resulting wave shape. The buttons let you choose one of four basic shapes: sinusoidal, triangular, rectangular, and randomly determined shapes. The **Skew** slider changes the basic waveform in different ways, depending on which waveform is chosen with the buttons, as follows:

- Sine: adds some extra bends to the wave
- Triangle: varies the wave from a downward-sloping ramp to an upward-sloping ramp
- Rectangle: changes the duty cycle of the rectangle (i.e. the relative length of the on and off segments)
- Random: adds ramping between the randomly chosen values

The small keyboard buttons affect whether Phosphor uses a separate LFO for each note. If the keyboard button is illuminated, each note in a chord will have its own LFO. Nothing synchronizes these separate LFOs, so they run independently and affect any modulation destinations independently in each note. If the keyboard button is dimmed, one LFO is shared by all notes and hence its effect will be synchronized between all held notes.

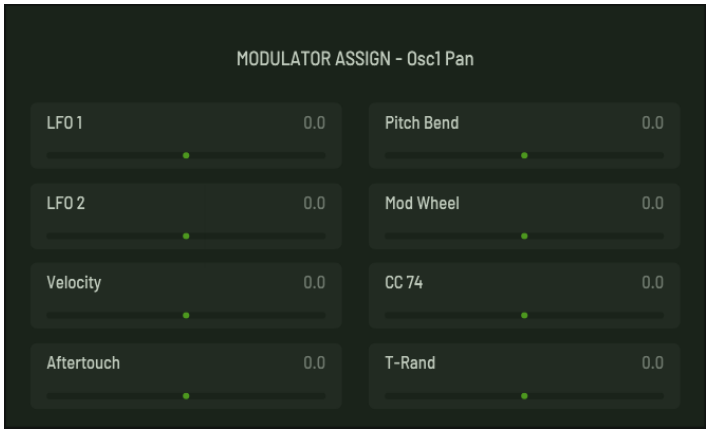
Finally, the **Level** slider controls the amplitude of the LFO's output signal. Obviously you'll have to set this slider somewhere above zero in order to hear the LFOs doing something. Using MIDI mod-wheel controller messages to modulate this slider lets you use the mod wheel to add vibrato or tremolo when you push it forward.

Modulation

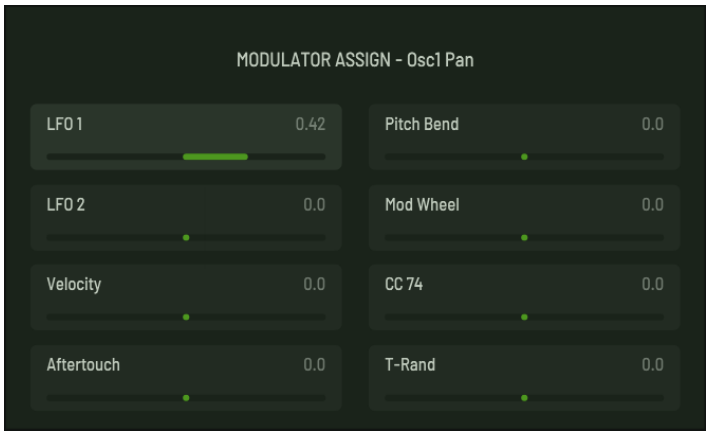
We've been waving our hands a little bit so far, mentioning modulation without explaining what we're talking about. Now we'll describe modulation in general and Phosphor's modulation system in particular. In the context of synthesizers, to *modulate* simply means to change something. If we speak of an LFO modulating a filter, we mean that the LFO changes the filter's frequency.

We also speak of modulation *sources* and *destinations*. A source is the thing that changes something else, and a destination is the thing that changes. In the LFO/filter example, the LFO is the source and the filter is the destination. Almost all of Phosphor's sliders, aside from the partial sliders, are modulation destinations. Phosphor's sources are the two LFOs, several types of MIDI messages, and a source of random numbers.

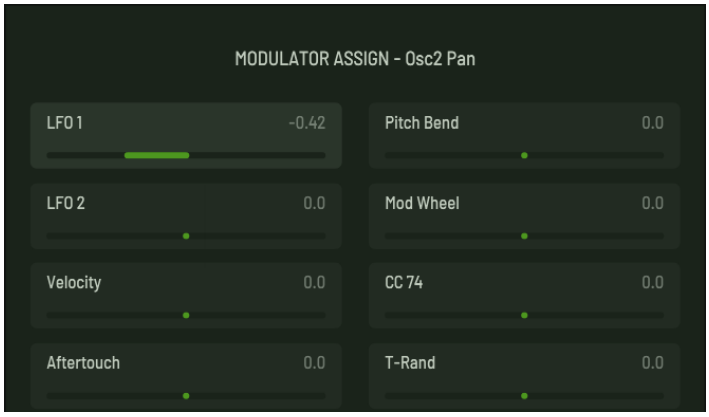
As an example, we'll use an LFO to modulate one of the voice's panning sliders, making the voice move back and forth in the stereo field, To make a connection between a source and a destination, first right-click on one of the sliders; for instance, the **Osc1 Pan** slider. This invokes the modulation routing panel:



Within the panel are sliders for each of Phosphor's modulation sources. The sliders are bidirectional. If a slider is moved to the right of center, the modulation source increases the destination; if a slider is left of center, it decreases the destination. The further the slider is from the center, the greater its effect on the destination. So, we might set our LFO1 slider a bit to the right to make the voice move back and forth:



Note that the output signals from the LFOs themselves are bipolar; that is, they vary between a positive value and a negative value. This means that the voice will move back and forth relative to the center of the stereo field, assuming that the **Osc1 Pan** slider is at its center. This also means that we can make the second voice move back and forth in the opposite direction by also modulating **Osc2 Pan** with LFO1, and setting the modulation amount to a negative value, inverting the LFO's signal:



To remove a modulation routing, set the slider to zero by double-clicking it. To dismiss the modulation panel, click anywhere outside of it.

The Pitch Bend, Mod Wheel, Velocity, Aftertouch, and CC74 modulation sources generate data from the most recently received MIDI note. The Pitch Bend and CC74 sources are bipolar, like the LFOs; the other sources are unipolar, generating only positive values. If MPE reception is enabled (see the Settings panel, described later), these MIDI messages arrive on individual channels and directed to individual voices.

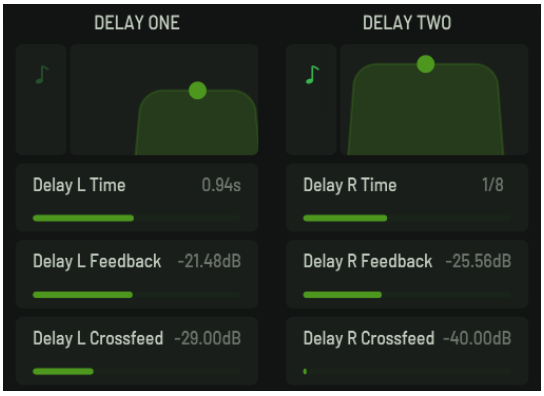
The **T-Rand**, short for Triggered Random, generates a bipolar random value every time Phosphor plays a new note.

When a slider receives modulation, a small yellow line appears above the slider. Its length and direction indicate the amount and polarity of the sum of the modulation sources, after they are scaled by the sliders in the modulation routing panel.

Delays and Filters

Because we at Audio Damage dearly love delay effects, we've added a stereo delay to Phosphor. The output of each voice passes through its own delay. Each delay has a range of 0 to 1999 msec, or approximately two seconds.

The **Delay Time** sliders operate either in units of time or in metrical units, that is, fractions of a beat. Click the musical-note buttons to switch between the two modes. If the note is illuminated, Phosphor uses the current tempo reported by your host to calculate its delay time. In this mode the delay time control sets the delay length in metrical units. The range of values is 1/32nd to 1/1 (a whole measure), with dotted and triplet times available. Triplet values are denoted with a "T" after the beat fraction, and dotted values are denoted with a period. For example, "1/8 D" indicates a delay time with a dotted eighth note feel. Phosphor will track tempo changes, saving you from having to adjust its delay time by hand when you change the tempo of your song.



The **Feedback** sliders control the feedback level, that is, the amount of the output signal which is fed from the output of the delay back into its input. Note that the feedback path goes through the filters.

The **Crossfeed** sliders control a second feedback path. As you move this slider to the right, some of the signal is fed to the input of the other delay. For example, if you move the **Delay R Crossfeed** slider, the right-channel output signal is fed back into the input of the left delay. This cross-channel feedback can be used to create delay effects that bounce back and forth.

Phosphor has low-pass and high-pass filters which come after the delay lines in the signal chain. These filters can be adjusted to alter the timbre of the delayed signal. For instance, you can use the low-pass filter to create a darker sound, or the high-pass filter to create a thinner sound.

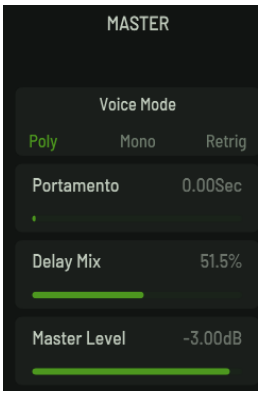
The filters are controlled in tandem with the graph-like controls. Since the filter controls change the low- and high-pass filters simultaneously the filters act like a band-pass filter with a variable width. Dragging the circle in the control changes the corner frequencies of both filters. Dragging up and down moves the corner frequencies farther apart and closer together; dragging left and right moves the frequencies in the same direction. As you pull the circle downwards, the filters move together and allow only a narrow band of frequencies to pass through. You can then move the dot left and right to hear that band move up and down in frequency.

Master Controls

The group of switches and sliders at the bottom right of Phosphor's window control Phosphor's response to MIDI keyboards and set output levels.

Voice Mode

The **Voice Mode** switches determines how Phosphor responds when you play more than one note at a time on your MIDI keyboard. There are three choices:



Poly – in this mode, Phosphor responds polyphonically, like a piano and most digital synthesizers. Play several keys on your keyboard, and you'll hear several notes.

Mono – in this mode, Phosphor plays only one note at a time. If you hold down a key on your keyboard, and press another key, Phosphor will not play a new note, but will change its pitch to that of the new key. In this mode, the envelope generators are not restarted when you play additional keys.

Retrig – this is the same as Mono mode, but the envelope generators are retriggered every time you press a key.

Portamento

The **Portamento** slider creates a sliding effect when the mono or retrig modes are active. If you hold down a key and press another key, Phosphor's pitch will slide to the second note at a rate determined by the slider. The further you move the slider to the right, the longer it takes Phosphor to slide from one note to the next. The Portamento slider has no effect if Poly mode is active.

Delay Mix

The **Delay Mix** slider controls the relative loudness of Phospor's raw output and the output from the delays and filters. If you set the slider to the far left, you won't hear the delayed/filtered signal at all. If you set it to the far right, you'll hear only the delayed/filtered signal. If you set it in the middle, you'll hear equal amounts of both.

Master Level

The **Master Level** slider controls Phosphor's overall loudness. Phosphor's signal can vary widely depending on how many partials you use and whether cross-modulation is employed; use this slider to compensate. Moving towards the right makes Phosphor's output louder.

Settings

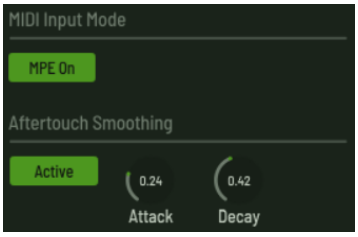
The Settings panel, invoked by clicking the small wrench icon near the top right of the window, contains controls for tailoring Phosphor to your MIDI controller and playing preferences. You'll also find controls for manipulating Phosphor's pitch standard and tuning. The settings in this panel are stored within your host DAW's session files, rather than within presets. Hence, they do not change when you load a preset file. These settings apply to individual instances of Phosphor; two or more instances within one DAW session can have different settings.

MIDI Input Mode

The **MPE On** switch, under the heading **MIDI Input Mode**, affects how Phosphor handles MIDI notes and controller messages. If you have a MPE controller such as a LinnStrument or Roli Seaboard, turn this switch on. Phosphor will then process pressure, pitch bend, and position information independently for each note, providing the expressivity that you expect from your controller. If you don't have a MPE controller, leave this switch turned off and ignore it.

As you're probably already aware, you also need a host program that can correctly transmit multi-channel MIDI information from your MPE controller to Phosphor for Phosphor to respond accordingly.

Aftertouch Smoothing



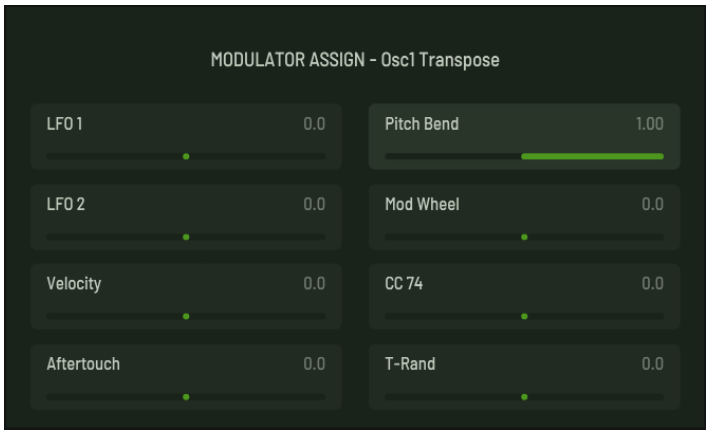
We used a variety of MPE-equipped controllers while developing and testing Phosphor. Most (maybe all) MPE controllers on the market provide their own software for adjusting and scaling their response to touch, pressure, position, and so on. We did not see a compelling reason to attempt to supplement or replace this functionality. However, we did find that it is often useful to smooth out the aftertouch (i.e. pressure) data originating from some controllers.

The **Active** button turns on a simple smoothing filter with a response rather like a lag processor module found in some analog synthesizers. The **Attack** and **Decay** knobs control the amount of smoothing: a higher setting produces more smoothing, that is, a slower change in output for a given change in input. The **Attack** knob affects increasing changes in value (more pressure) while the **Decay** knob affects decreasing changes (less pressure).

Some controllers do not send a zero aftertouch value when you remove your finger from the key or pad. If Phosphor's aftertouch smoother is active, the aftertouch modulation source will always return to zero after a note ends. How quickly it fades to zero from its current value depends on the setting of the Decay knob.

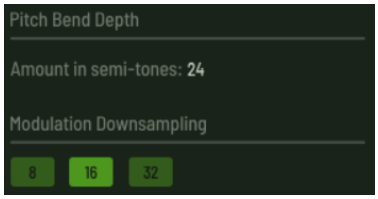
Pitch Bend Depth

Phosphor's modulation system allows MIDI pitch bend messages to control any parameter with any range of effect. The **Pitch Bend Depth** setting changes the maximum modulation value for the pitch-bend signal. This value is in semitones; click and drag vertically to change it or double-click it to type a value. The amount slider of any modulation connection which uses MIDI pitch bend will be affected by this value. To achieve the usual action for the pitch-bend controller—changing the pitch of the oscillators—modulate the Osc 1 Transpose and Osc 2 Transpose with Pitch Bend, with the modulation slider set all the way to the right, like this:



(Of course, nothing prevents you from setting the modulation amounts to different values, or one to a negative value so that the oscillators respond differently to the pitch bend control.)

Modulation Downsampling



To reduce Phosphor's overall CPU load, its modulators operate more slowly than the rest of the signal processing, doing the math to generate and route the modulation signals less often than every audio sample. This reduction in the number-crunching load means a slight reduction in the smoothness of the modulation, which can sometimes affect the audio signal. The three buttons labeled **8**, **16**, and **32** let you choose how often modulation is updated, with the number representing how many audio samples are calculated before the next modulation calculation happens. In other words, clicking the **16** button means that Phosphor's modulators will be updated every 16 samples. Usually this setting won't have much effect and you can leave it at 32, which provides the lowest CPU load. Try the other settings and let your ears be your guide. You may hear a difference in bass sounds with their relatively slow audio waveforms, or if you're using one modulator to affect another or itself.

Global Tuning

Phosphor provides facilities for non-standard tuning and intonation. First, the **A4 Frequency** control simply adjusts Phosphor's overall pitch. Standard tuning uses 440Hz for A above middle C. If you need to adjust Phosphor's pitch slightly to match other instruments or recordings, or if you use a different pitch standard in your music, simply change the number in the text box, either by clicking and dragging vertically or double-clicking and typing a number.

Second, Phosphor's pitch reference, intonation, and/or keyboard mapping can be completely changed with TUN files. TUN files use a standard file format for providing tuning information. If you're interested in microtonal music or non-Western musical instruments, you've probably already heard of them. If you're familiar with Scala but not TUN files, Scala can export TUN files. One TUN file contains the information that Scala places in separate SCL and KBM files. A web search will reveal plenty of information about creating TUN files.

To load a TUN file, click on the text field to the right of the word **Table**. Phosphor will prompt you with a standard file dialog box; choose your file and Phosphor will adopt the tuning and mapping found in that file. Once a file is loaded, Phosphor keeps a reference to that file within its settings. If you move the TUN file, you'll have to re-load it into Phosphor.

Once a TUN file is loaded, the **A4 Frequency** control is disabled and cannot be changed. The pitch reference in the TUN file always overrides the A4 frequency setting.

To revert to the usual 12-note equal-tempered tuning, click the **Clear** button.

Startup Settings

Once you've tuned the settings to your liking, clicking the **Save** button under Startup Settings. Phosphor will save the current settings in its preset folder in a new file named `globals.cont`. Every new instance of Phosphor will load its settings from this file.

iOS Note

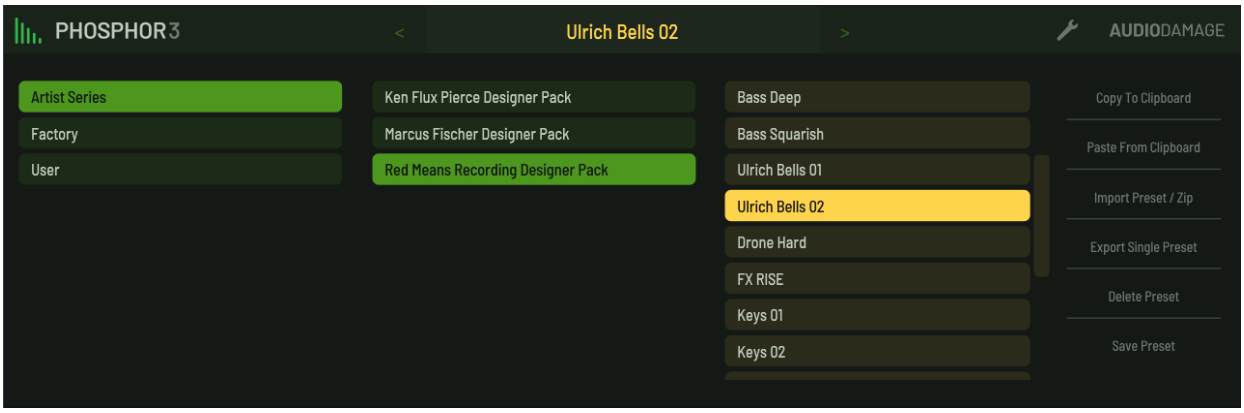
Because of the vagaries of the iOS file system, Phosphor saves a copy of TUN file when you click the **Save** button under Startup Settings. Phosphor then loads that copy automatically along with other saved settings. If you change the original TUN file, you need to manually reload it into Phosphor.

Audio/MIDI Settings

On iOS only, clicking this button invokes a window for setting the audio I/O and MIDI options for the standalone version of Phosphor.

Presets

Phosphor includes a number of presets to serve as a demonstration of its capabilities and inspirations for your own creations. To access the presets, click the name of the current preset at the top center of the window to open the preset browser. Click anywhere outside the preset browser to dismiss it.



The browser displays presets within lists of folders. The leftmost list shows the folders within Phosphor's preset collection, grouped in three categories: Artist Series, Factor, and User. Clicking any of these folders reveals its contents in the middle list. The Artist Series and Factory folders contain sub-folders of their own; clicking any of these displays their contents—the presets—in the rightmost list. Clicking on a preset name loads the settings into Phosphor.

After clicking on any of the three lists, you can move up and down in the list with the corresponding arrow keys on your keyboard. You can also go through all the presets in all the folders by clicking the left and right pointers on either side of the preset name at the top center of Phosphor's window.

Loading a preset irretrievably erases Phosphor's current settings, so if you have created a sound that you want to use again, save it as a new preset before loading another preset. To save your own presets, click the **Save Preset** button at the right edge of the window. Phosphor will prompt you to enter a name for the preset with a standard system file dialog box.

The folders and presets in the browser correspond to folders and files within Phosphor's own folder on your storage device (i.e. your computer's hard drive or SSD). This folder is located at `C:\ProgramData\Audio Damage\Phosphor3\` on Windows, and `~/Music/Audio Damage/Phosphor3/` on macOS. Theoretically you can save your presets anywhere you like, but for them to show up in Phosphor's User list they must be placed in the User folder within Phosphor's folder. Also, to avoid possible collisions during future updates, do not store your presets within the Artist Series or Factory folders.

Any folders you create within the User folder will show up as folders in the User list. Folders within sub-folders of the User folder are not available within the preset browser, so don't get carried away with nested folders.

Preset files are plain-text XML files so that you can exchange them online in forums, copy them between a Windows computer and a Macintosh, email them to your friends, etc.

You can delete presets and folders from the lists by clicking their name and then clicking the **Delete Preset** or **Delete Folder** button. Phosphor will give you a chance to confirm this action or cancel it. If you confirm, the preset/folder will be removed from your storage system and is gone for good.

Importing and Exporting Presets

The **Copy To Clipboard** and **Paste From Clipboard** buttons copy Phosphor's current settings to the system clipboard and paste settings from the clipboard. You can use the copy and paste commands to transfer settings between two instances of Phosphor or paste the settings into an email message or text editor. When copied to the clipboard, presets are presented in the same XML text as used in preset files.

The **Import Preset / Zip** button provides a way to add presets to Phosphor without manually moving them into the appropriate folders in your file system. Clicking this button produces a file-browser window wherein you can select either a single preset file or a .zip file containing one or more presets. After you select the file, Phosphor copies the preset(s) into whichever folder you have selected in Phosphor's preset list, unzipping the file first if necessary.

Depending on whether you've selected a preset or Folder, the **Export Single Preset** or **Export Folder As Zip** button performs the complementary functions of the **Import** button. First select either a preset or a folder in Phosphor's list, then click the export button. A file-save window appears; choose a location in your file system, give the file a name, and click Save. If you have chosen a folder in Phosphor's preset list, the plugin places it and all of the presets it contains in a .zip file.

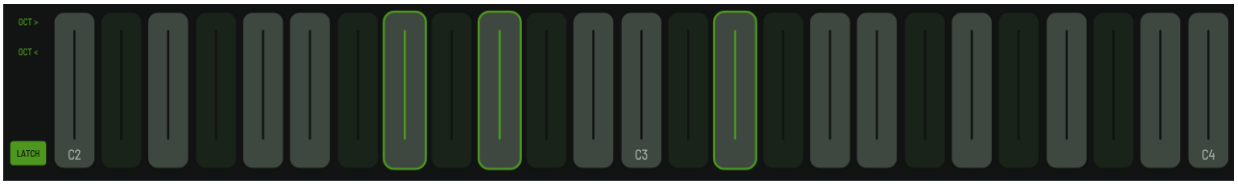
Default Preset

If you save a preset with the special name “Default” in the User folder, new instances of Phosphor will load it automatically when you add it to your DAW session. You can use a default preset file to give you the same starting point with Phosphor, maybe with a few modulation settings that you always use, maybe a complete sound that you find yourself using on every new song in your current project.

Keyboard

At the bottom of Phosphor’s window, you’ll see a musical keyboard. Particularly useful for the iOS version, a few moments of experimentation should reveal everything you need to know about it. However, for the sake of completeness, we’ll go over its features here.

While all the keys are the same shape and size, the keyboard follows the traditional layout of a piano keyboard. Small labels indicate the C keys and their octave. You can shift the keyboard up and down by octaves with the OCT buttons at the left.



While touchscreens are not (yet) sensitive to pressure, Phosphor’s keyboard generates velocity information based on the position of your finger or mouse click. Tapping nearer the top end of the keys sends higher velocity data. This data appears as Velocity in Phosphor’s list of modulation sources; the keyboard does not transmit MIDI information externally. You can also slide your finger or mouse horizontally to bend the pitch of the notes.

Turning on the Latch button cases keys to remain held after you release them. Tapping a latched key a second time turns it off. Use the Latch button for creating drones, or for freeing a hand to adjust the sliders, etc.

Automation

Most of Phosphor’s parameters, other than the partials sliders and mode switches, can be automated using your host's automation features. Consult your host's documentation for information on how to use these features.

And Finally...

Thanks again for purchasing Phosphor. We make every effort to ensure your satisfaction with our products and want you to be happy with your purchase. Please write to support@audiodamage.com if you have any questions or comments.