OPERATION MANUAL

CYPHER2

fxpansion

Operation Manual

1 Cypher2 Introduction

Cypher2

Programming: Angus Hewlett
Interface design: Angus Hewlett, Raimondo Taibi, Yuta Sugawara, Rory Dow
Sound design: Rafael Szaban, Rory Dow, Mayur Maha, Lawrence King, Adam Smith and Nico Jaussaud
Documentation: Mayur Maha
Videos: Rory Dow, Lily Skove, Heen-Wah Wai
Artwork / Photography: Rus Brockman, Jenny Tillyer
Project Management: Rhiannon McLaren
QA: Lawrence King, Lewis Gunson, Jared Fountain
Marketing: Will MacNamara, Jack Sutton

With thanks to:

SKoT McDonald, Drew Vernon, Tom Meaney, Mike Bugh, Paul Chana, Steve Baker, Andy Simper, Richard Sven Patterson, Tom Maisey, Manon Dave, Adrian Dybowski, Geoffrey Gee, Hollin Jones, Loudon Stearns, Laurent de Soras, Antti Huovilainen and the entire ROLI team

Operation Manual revision 2

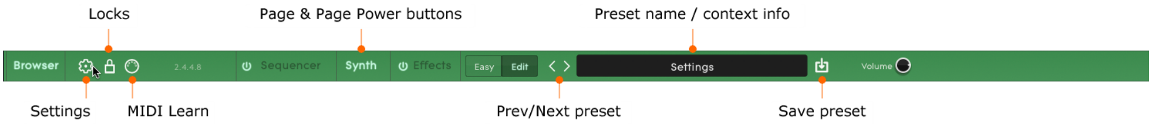
1.1 Interface overview

Note: if Cypher2 is currently in Easy mode ('preset-player' mode), click the **Edit** button to display the full interface.

Browser

The Browser provides 1-click access to Cypher2's library of presets. It also allows presets created or modified by the user to be saved and loaded.

Navigation bar



Browser

The Browser's visibility can be toggled by clicking the **Browser** button above the Browser area.

Page & Page Power ⏻ buttons

The **Sequencer Page**, **Synth Page** and **Effects Page** buttons each switch the Main editing area to a different 'page' of operation:

- **Synth page**

Synthesis engine

The Synth page contains the synthesis engine parameters for Cypher2 - all parts of the audio path before the effects section - as well as the Scope panel which contains the Visualizer - a context-sensitive graphical display for various parts of the synthesis engine.

Modulation

The Synth page also includes several sections used for modulation: 2 Dual LFOs, 2 Ramp Generators and 3 Envelopes.

In addition, the Scope panel can be switched to display the Euclid, Curve Processor, Maths and KeyZone editors, used for more advanced modulation processing.

• **Sequencer page**

This page contains Cypher2's built-in Step Sequencer and Arpeggiator. The **Sequencer Page Power** button in the Navigation bar activates/ deactivates the Sequencer functionality entirely.

• **Effects page**

The Effects page contains 2 FX chains (FX A and FX B), each of which features 3 slots in which to insert FX devices. 30 built-in FX devices of various kinds are available. The **Effects Page Power** button toggles the entire Effects section on or off.

Easy / Edit mode

These buttons switch between Easy mode and Edit mode. **Easy** mode provides no means to edit sounds aside from adjusting macro controls and is designed for performance situations in which editing is not required. **Edit** mode provides full editing functionality.

Previous / Next preset

The **Previous** and **Next** buttons cycle through the available presets in the Browser listing sequentially.

Preset name / context info

This display performs 2 functions: it displays the currently-loaded preset name but also shows context-sensitive information when the cursor is positioned over a control.

Save preset

Click this button to display the Save preset dialog, allowing a preset name and several description fields to be edited before saving.

Master Vol (Volume)

This control adjusts the final output level after the Effects stages, allowing the volume to be boosted by up to 12dB or attenuated to -∞ dB.

Settings

This button displays the Settings menu allowing various preferences to be adjusted.

Locks

This button displays the Locks menu allowing groups of controls from being affected indirectly such as by preset-switching, morphing or randomizing processes.

MIDI Learn

Activating the **MIDI Learn** button initiates MIDI Learn mode for assigning Cypher2 controls to MIDI CC controller messages.

Global synth controls

This section of the interface is always visible, even when viewing the Sequencer or Effects pages.

These settings relate to the way that Cypher2 responds to being played - including polyphony, unison, note priority, master tuning, pitch bend and glide functionality.

This area also contains the **P1** (Perf1), **P2** (Perf2) and **P3** (Perf3) macro performance controls which provide TransMod-based modulation from real-time controller input. The performance controls are designed to be assigned to available hardware MIDI CC controllers using the MIDI Learn function.

TransMod modulation slots

The 24 TransMod modulation slots are central to Cypher2's advanced but intuitive TransMod modulation system.

Although Cypher2 provides dedicated controls for some modulation (oscillator/filter keyboard tracking and filter envelope modulation), the TransMod modulation system goes far beyond these capabilities.

The upper 16 TransMod slots can modulate virtually all Cypher2 parameters from a single modulation source, optionally scaled (multiplied) by another modulation source.

8 additional single-source slots are provided with hard-wired sources.

Quick-presets, Morphing and Randomizer

This section allows 8 presets to be loaded into the available Quick-preset slots for fast preset-switching and morphing. Morphing can be used for radical timbral shifting and 'frozen' to create new sounds.

Randomization works in a similar way - parameters change within a variable range, within which the sound can be frozen as a new preset.

1.2 Adjusting sliders and rotary controls

Adjusting initial values of parameters (No TransMod slot selected)

Sliders

Move the cursor above the slider cap. Notice that it is highlighted. Now either:

• use the mousewheel or trackpad scroll function

• or click and drag the slider cap up/down.

The new value is shown in real time on the tooltip which appears.

• It is also possible to double-click the slider cap, type a new value in the box which appears and press ENTER or RETURN.

Rotary pots

Move the cursor above the rotary control. Notice that it is highlighted. Now either:

• use the mousewheel or trackpad scroll function

• or click and drag the control up/down.

The new value is shown in real time on the tooltip which appears.

• It is also possible to double-click the control, type a new value in the box which appears and press ENTER or RETURN.

Fine control over parameters

Hold down the SHIFT key while adjusting a control for finer resolution.

1.3 Adjusting TransMod modulation

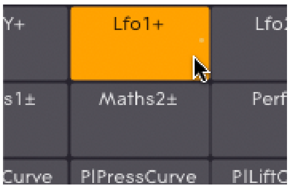
Introduction to TransMod modulation

The TransMod system provides an intuitive but very deep modulation system for Cypher2's parameters. See chapter 5 for a full guide to understanding the TransMod system.

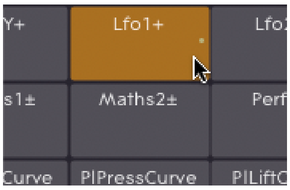
Selecting TransMod slots

To begin using TransMod modulation, one of the available TransMod slots must first be selected. Each TransMod slot can be assigned to a modulation source which is chosen from an extensive list of monophonic (applied to all playing notes) and polyphonic (applied to each note individually) modulation sources.

Selecting a TransMod slot

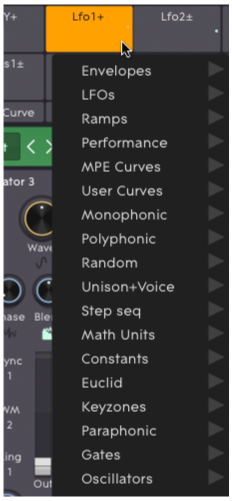


• to select a slot without displaying any menus, click anywhere within the slot.



• to deselect the slot, click it again.

• **right-click** on the *upper* part of the slot to select a slot and display its TransMod *Source* menu with category sub-menus containing a variety of sources.



• **right-click** on the *lower* part of the slot to select a slot and display its TransMod *Scale* menu - a multiplier for the source specified in the upper part.

The vast range of TransMod modulation sources (which can also be used as 'Scale' functions which multiply a slot's modulation source) are described later in this manual.

When a TransMod slot is selected, TransMod-enabled parameters in Cypher2 display and allow editing for modulation depths. The vast majority of parameters in Cypher2's Synth, Effects and Sequencer pages can be modulated. However, some controls such as buttons and certain other parameters cannot be modulated.

Adjusting TransMod modulation depths: Sliders

When a TransMod slot is selected, Cypher2's slider-type controls can be manipulated in the following ways:

Creating modulation on sliders

Position the cursor within the slider but *above or below the slider cap*. Notice that the entire slider is highlighted. Now either:

• use the mousewheel or trackpad scroll function

• or click and drag up/down.

The initial value and new modulation value are both shown in real time on the tooltip which appears.

• It is also possible to double-click above/below the slider cap, type a new value in the box which appears and press ENTER or RETURN.

Adjusting the initial value with or without modulation

Move the cursor above the slider cap - notice it is highlighted.

• Use the mousewheel to adjust the initial value - the modulation depth is preserved.

• or click and drag up/down.

The initial value is shown in real time on the tooltip which appears.

• Alternatively, double-click and type a new value.

• It is also possible to hold down the ALT key while clicking and dragging the slider cap (or scrolling with the mousewheel/trackpad) to adjust the initial value *without* preserving the modulation depth.

Adjusting TransMod modulation depths: Rotary controls

Creating modulation on rotary controls

Move the cursor above the *outer ring* of the rotary control. Notice the highlighting around the control. Now either:

• use the mousewheel or trackpad scroll

• or click and drag up/down.

The initial value and new modulation value are both shown in real time on the tooltip which appears.

• It is also possible to double-click in the outer ring, type a new value in the box which appears and press ENTER/RETURN.

Adjusting the initial value with or without modulation

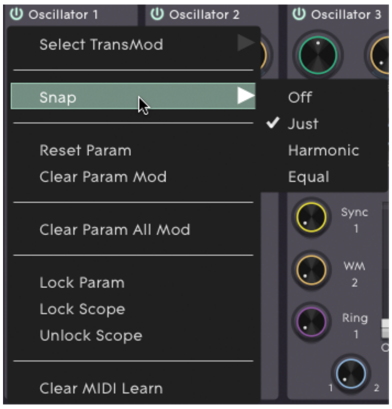
Move the cursor above the main part of the rotary control and either:

- Use the mousewheel
- Click and drag up/down
- Double-click and type a new value

• It is also possible to hold down ALT while clicking and dragging the main part of the rotary control (or scrolling with the mousewheel/trackpad) to adjust the initial value *without* preserving the modulation.

1.4 Unit and Snapping modes

Snapping and Unit modes



Controls related to tuning oscillator pitch or filter cutoff frequency offer 4 distinct modes of operation which are accessed via the parameter context menu.

The *Just* and *Harmonic* modes are based on perfect pitch ratios as units, rather than absolute frequency settings.

These modes persist independently for each control. Note that the **Fine** oscillator pitch and **Detune** controls are not affected by snapping modes, which are designed purely for tuning the main oscillator and filter pitch (the latter being especially useful when using self-oscillation at high resonance).

Off

In the *Off* mode, all snapping is deactivated - the control is set in semitones but does not snap to whole semitones.

Just

Just mode uses perfect pitch ratios as units rather than imperfect, equal-tempered pitch. In this mode, pitch or filter cutoff controls are set in harmonics (Hm). There is no snapping to whole harmonics.

Harmonic

The *Harmonic* mode is similar to *Just*, except that additional snapping to whole harmonics occurs (although SHIFT can be held down for fine control).

Equal

In *Equal* mode controls are set in equal-tempered tuning in semitones and snapping to whole semitones occurs (although SHIFT can be held down for fine control).

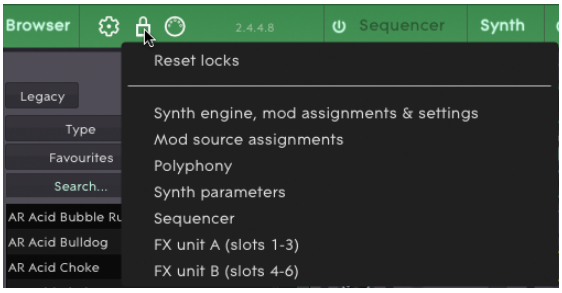
1.5 Locks

Locks menu

Cypher2's Locks allow various parts of the state of Cypher2 to remain unaffected when loading new presets.

Locks are also very useful when using Quick-preset morphing or randomizing to ensure that certain parameters are unaffected by changes.

• Please note that when using Locks, presets may sound very different to how they are intended to sound!



Any currently locked elements are highlighted in red.

Click the **Locks** button above the Browser area to display a menu showing the various Locks that can be activated or deactivated:

Reset locks

This function removes all previously activated Locks.

Synth engine, mod assignments & settings

All synth parameters and modulation depths, TransMod slot Source and Scale function assignments and the following additional controls are locked:

- **Voices**
- **Unison**
- **Bend Up/Bend Down**

Mod source assignments

TransMod slot source and scale function assignments are locked.

Polyphony

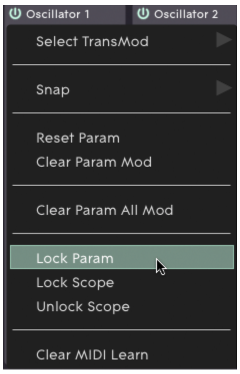
The **Voices** and **Unison** settings are locked.

Synth parameters

All synth parameters and modulation depths are locked.

Sequencer

The content of the Sequencer page is locked.



FX unit A (slots 1-3)

The content of the Effects page's FX A chain is locked.

FX unit B (slots 4-6)

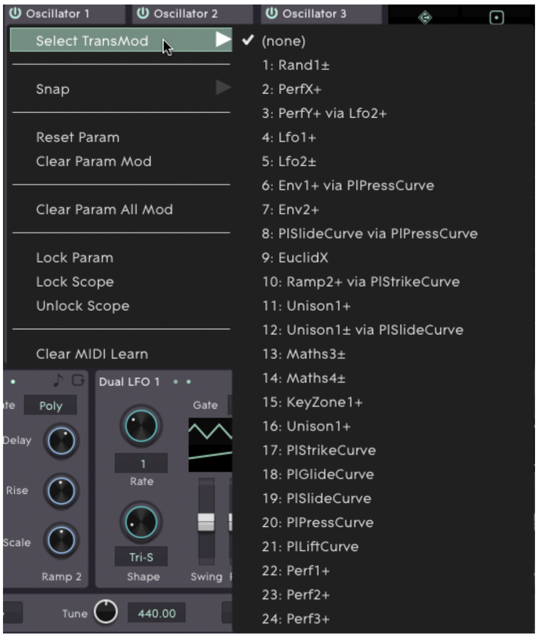
The content of the Effects page's FX B chain is locked.

Locking individual parameters / TransMod slots

Individual controls can be locked using the Parameter context menu - right-click on a parameter to display this menu, then use the **Lock Param** function. Similarly, right-click on a TransMod slot - the menu features a **Lock** function for the slot.

1.6 Parameter context menu

This menu is displayed by right-clicking (or using any other secondary click method) on a parameter or control in Cypher2.



Select TransMod

This sub-menu provides an alternative means of selecting a TransMod modulation slot without having to move the cursor away from a control in the midst of sound design: right-click the parameter, select the desired TransMod slot from the menu and then create TransMod modulation depths on the control (and on any other controls) as desired.

Snap

The **Snap** sub-menu is available for pitch-based parameters and specifies the current Unit/Snapping modes.

Reset Param

This function sets the parameter to its default value.

Clear Param Mod

Use this function to clear any modulation depth that exists for the parameter from the *current TransMod slot*.

Clear Param All Mod

This function clears any modulation depths that exist for the parameter from *all TransMod slots*.

Lock Param

When activated, the **Lock** function protects the parameter from being affected during preset-loading or Quick-preset switching/morphing/randomizer operations. Entire sections of parameters can also be locked using the Lock menu.

Lock Scope

This function is used to lock the Scope's context-sensitive Visualizer display for the section within which the parameter is located.

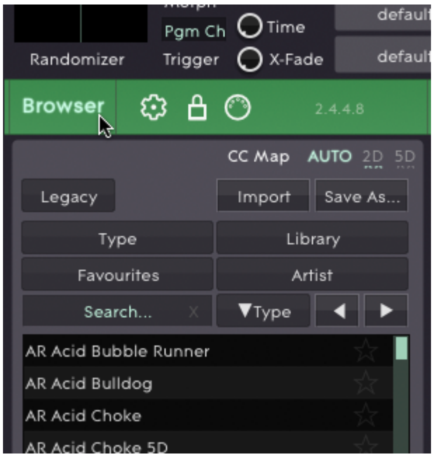
Clear MIDI Learn

Use the **Clear MIDI Learn** function to unassign the parameter from any MIDI Learn mapping.

2 Browsing presets and Easy mode

Cypher2's Browser provides quick access to all factory and user presets. It also provides several additional functions for managing presets and searching/filtering based on various criteria.

Show/hide Browser

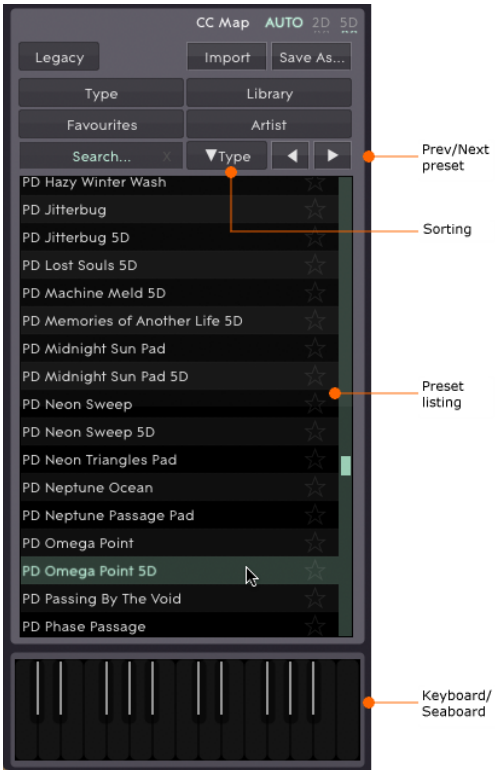


The Browser can be shown or hidden as needed.

To hide the Browser if it is visible, click the **Browser** button at the far left of the Navigation bar.

To display the Browser if it is not currently visible, click the **Browser** button again.

Loading and saving Presets



Preset listing

The main part of the Browser area shows the list of available presets. By default all factory-supplied and user-created presets are shown although the list can be narrowed down using the Browser's various filtering and search functions.

- Click any preset in the list to load it.
- To revert to the previous state of Cypher2 before loading the preset, click the **Cancel** button which appears when a preset is first loaded, before it is subsequently modified.

Prev / Next preset ◂ ▸

Cycle through the available presets sequentially using the **Previous** ◂ and **Next** ▸ buttons.

It is also possible to cycle through presets using keyboard cursor-key shortcuts. Note that some DAW/host applications may intercept keyboard input meaning these cannot be used.

↑	Previous preset
↓	Next preset

Import

The **Import** function can load a preset from any location and optionally add it to Strobe's preset database. This may be more convenient than manually copying the preset to the required folder.

Cancel

The **Cancel** button is displayed after a preset is loaded, before it is subsequently modified. It returns to the state of Cypher2 before the preset was loaded - this is useful if a new preset is accidentally loaded.

Save As

Click the **Save As** button to save the current state of Cypher2 as a preset, specifying a name and various description fields. The saved preset is placed in the user presets location.

Quick-preset slots

When a preset is loaded from the Browser, it is loaded into the current Quick-preset slot (of which there are 8 available).

CC Map controls

As well as the various preset browsing controls, the Browser also contains the **CC Map** controls - these are used in conjunction with the MIDI Learn function.

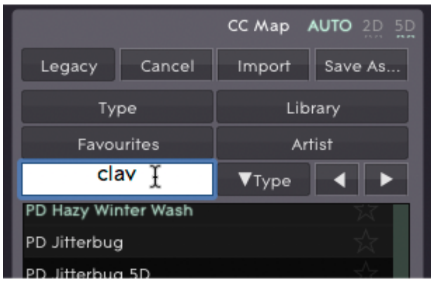
Keyboard / Seaboard

The on-screen keyboard allows notes to be played with no hardware keyboard available, useful for verifying that Cypher2 is operating correctly. Note that the keyboard changes between a 'piano'-style keyboard view and a Seaboard-style MPE controller view depending on the currently selected **CC Map**.

Searching, Sorting and Filtering

The Browser contains a number of additional features for searching or filtering the available presets to make it easier to find sounds.

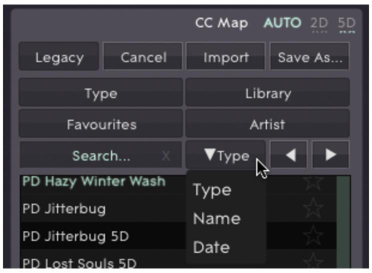
Search



Double-click the **Search** box and enter one or more terms to show only presets whose names match the search term criteria.

The search results are displayed in the Preset listing area until the **Cancel search** button is clicked.

Sorting

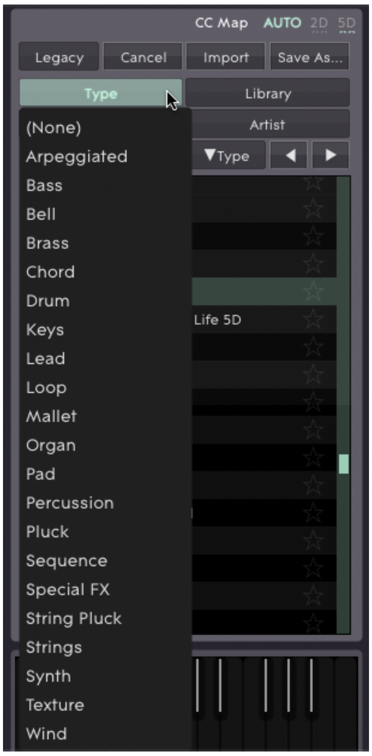


The **Sorting** drop-down menu to the right of the Search box allows the preset listing to be listed by *Type*, by *Name* or by the *Date* it was last saved.

Sorting by *Date* can be very useful while creating your own sounds, as the most recently saved presets are shown at the top of the list.

Filters

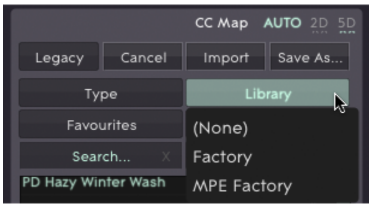
The Browser's Filters provide drop-down menus which allow filtering of the preset list according to various criteria.



Type

The **Type** drop-down menu allows the preset list to be filtered to show only 1 of the available preset types. Select *(None)* for the filter to show all preset types. The following preset types are available:

- *(None)*
- *Arpeggiated* (**AR**)
- *Bass* (**BA**)
- *Bell* (**BL**)
- *Brass* (**BR**)
- *Chord* (**CH**)
- *Drum* (**DR**)
- *Keys* (**KB**)
- *Lead* (**LD**)
- *Loop* (**LP**)
- *Mallet* (**MT**)
- *Organ* (**OR**)
- *Pad* (**PD**)
- *Percussion* (**PE**)
- *Pluck* (**PL**)
- *Sequence* (**SQ**)
- *Special FX* (**FX**)
- *String Pluck* (**PS**)
- *Strings* (**ST**)
- *Synth* (**SY**)
- *Texture* (**TX**)
- *Wind* (**WI**)



Library

The **Library** filter allows either *Factory* or *User* presets to be exclusively visible. To show both Factory and User presets in the list, select *(None)* for the filter.

Artist

The **Artist** filter allows presets by a single preset author to be seen. To show presets by all preset designers, select *(None)* for this filter.

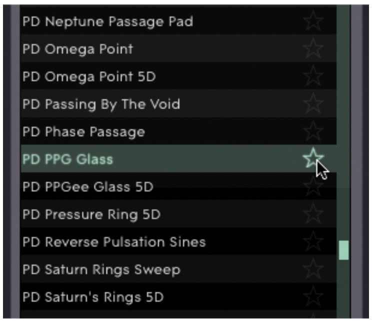
Other controls

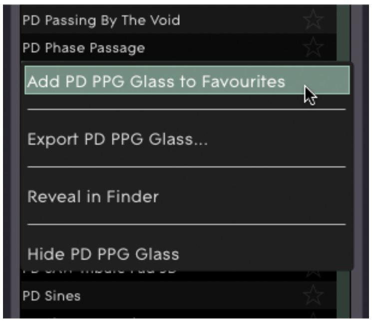
The following buttons toggle between 2 distinct preset views. These views can be combined with search and filter criteria.

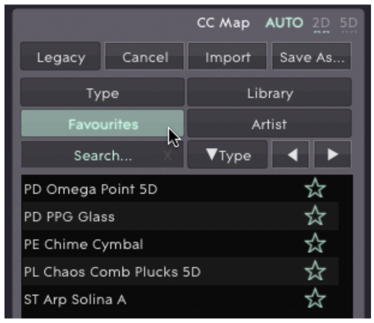
Legacy

Activating the **Legacy** button displays only legacy Cypher v1.x presets, if DCAM Synth Squad is present on the system on which Cypher2 is installed. Deactivate the button to show only Cypher2 presets.

Favourites







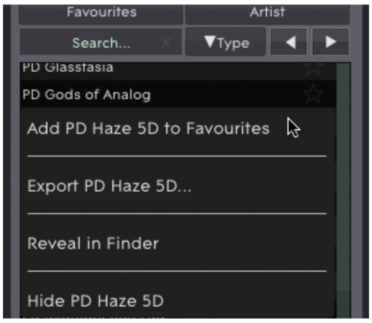
• To mark a preset as a Favourite, activate the **Favourite** ✩ button to the right of the preset name in the preset list by clicking it.

• Alternatively, use the **Add preset to Favourites** function in the Browser context menu (right-click on a preset to display this menu).

• Activating the **Favourites** button displays only presets which have been previously marked as Favourites.

To remove the Favourite status for a preset, deactivate the **Favourite** ✩ button by clicking it again or use the **Remove preset from Favourites** function in the Browser context menu.

Browser context menu



Right-click on any preset in the preset list to display the Browser context menu.

Add preset to Favourites
Remove preset from Favourites

These functions are described above.

Export preset...

This function exports the preset to any location on disk. This is useful for sharing a preset without having to navigate to the relevant location on disk to find it.

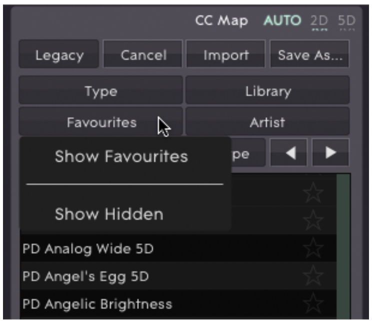
Reveal in Finder... (Mac)
Reveal in Explorer... (Windows)

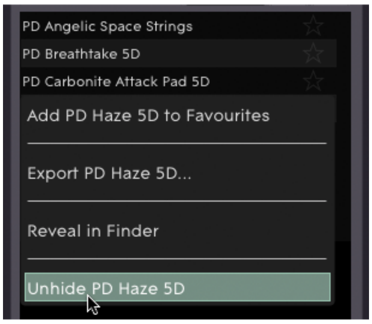
This function opens a system file browser window to show the factory or user folder which contains the preset.

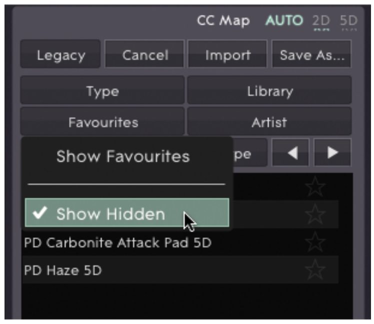
Hide preset
Unhide preset

The Browser provides the ability to hide presets, which can be considered as an inverse counterpart to the Favourites system described above.

To hide a preset from the preset list, use the **Hide preset** function on the Browser context menu. Hidden presets are not deleted - they can still be viewed and subsequently unhidden whenever required.







• To view all currently hidden presets, right-click the **Favourites** button and use the **Show Hidden** function on the menu that appears.

• To remove the Hidden status from any preset, right-click on the preset and use the **Unhide preset** function.

• Right-click on the **Favourites** button and click the **Show Hidden** function again to return to the main Preset listing (minus any Hidden presets).

Factory and User preset locations

• **Factory presets** are stored within the following locations:

Mac OSX: /Library/Application Support/FXpansion/Cypher2/Presets
/Library/Application Support/FXpansion/Cypher2/Collections

Windows: C:\Program Files (x86)\FXpansion\Cypher2\Presets
C:\Program Files (x86)\FXpansion\Cypher2\Collections

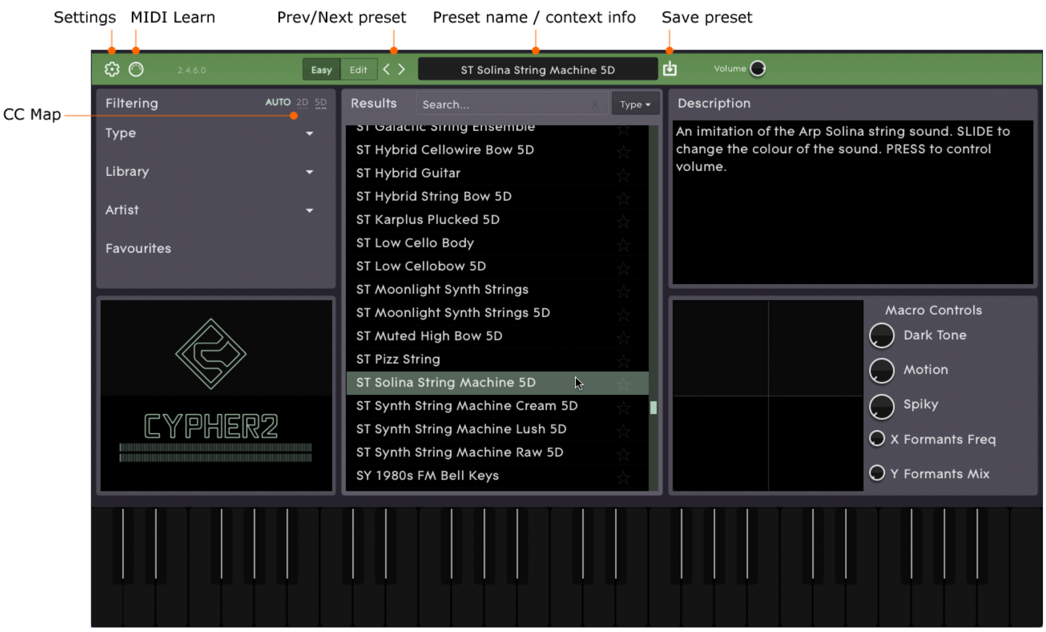
• **User presets** are stored within the following locations:

Mac OSX: /Users/<username>/Documents/FXpansion/Cypher2/Presets

Windows: C:\Users\[Username]\Documents\FXpansion\Cypher2\Presets

Sub-folders within the factory and user preset folders are used as Artist (preset author) names for the presets within them - use the **Artist** filter to show only the presets within any sub-folder.

Easy mode



Easy mode provides a 'preset player' mode simply for browsing and playing Cypher2 presets, also exposing the macro controls (P1, P2, P3 and X-Y pad) for easy manipulation of sounds as long as presets contain suitable macro/X-Y pad assignments.

When Easy mode is active, the Cypher2 Synthesis, Sequencer and Effects page cannot be accessed. These are only available in *Edit mode*.

If Edit mode is active, click the **Easy** button to display the interface in Easy mode. To return to Edit mode, click the **Edit** button.

Settings
Save...
Volume
Preset name / context info
Previous / Next Preset

These functions are identical to those in the Edit mode's Navigation bar.

Edit mode

Click the **Edit** button to switch the interface from Easy mode to Edit mode.

MIDI Learn & CC Map

These controls are used with the MIDI Learn function.

Filtering: Type / Library / Artist / Favourites
Search
Sorting
Results (Preset listing)
Keyboard / Seaboard

These functions are identical to the corresponding functions in Cypher2's Edit mode Browser - see above.

Description

This section displays the preset description which can optionally be saved along with the preset in the Save dialog.

Macro controls

This part of the Easy page provides controls for the 5 available performance macros: **P1**, **P2**, **P3** and **Base X** / **Base Y** in the Euclid processor. Like the preset's **Description**, the names of macros can be specified when saving a preset.

3 Synth page

This chapter describes Cypher2's synthesis sections:

Oscillator Section

Cypher2's oscillator section is central to its timbral range. It includes a wide variety of audio-rate modulation functions for the 3 oscillators, which feature waveforms which continuously morph from triangle/sine to saw to variable-width pulse.

Each oscillator features a routing control which sets the balance of its output within the dual Shaper and Filter paths.

Shaper and Filter sections

Cypher2's dual Shaper / Filter sections can be considered as the tone-shaping part of the voice, in which the harmonically complex output from the oscillators is sculpted by circuit-modelled tone modifiers.

The Shaper sections allow harmonics to be added to the signal: the Fold types in particular can add very interesting overtones which, when applied to a 'simple' source such as a sine, can resemble the effect of a low-pass filter opening on a harmonically complex source.

The Filter sections represent the 'subtractive' part of the term 'subtractive synthesis' - it subtracts parts of the frequency spectrum from the oscillator signal.

Output Mixer - Amplifier section

This section provides the final V.C.A. which 'articulates' the amplitude of each synth voice. The oscillator is always running internally - the V.C.A. uses a control signal from an envelope to shape the amplitude (or 'loudness') of each voice when notes are played.

The Output section actually provides a quad-stereo V.C.A. - the signals from 4 'taps' at various points in the signal path are available: the signal before and after each Shaper - Filter path. All these 'taps' are affected by the **Pan** control and are provided as a stereo output fed into the Effects section.

Analogue Noise section

The Analogue Noise section simulates the effect of noise and mains hum in certain parts of the audio and control signal paths, something that usually occurs in real analogue synth circuits to some extent.

Scope / Visualizer

The multi-tab Scope/Visualizer panel performs a number of functions in Cypher2.

With the **Visualizer** button activated, it provides graphical feedback when the cursor is moved above the various parts of the synth engine or used to adjust certain controls. It should not be regarded as an 'oscilloscope' but rather as an aid to visualising each part of the synth and current signal routings.

Other functions on the Synth page

This chapter covers only functions directly related to the audio synthesis engine. The Synth page also contains a number of editing functions for modulators in Cypher2, which are described in the Modulation chapter. All these modulators, except where stated, must be routed to modulate parameters in Cypher2 using the TransMod system.

Gated modulators

The Ramp, Dual LFO and Envelope modules must be used with the TransMod system to modulate most Cypher2 parameters although the following direct modulation routings also exist:

Envelopes 1 & 2 feature direct routings to modulate **Link**-activated Filter sections but can also be routed to other parameters in Cypher2 via the TransMod system.

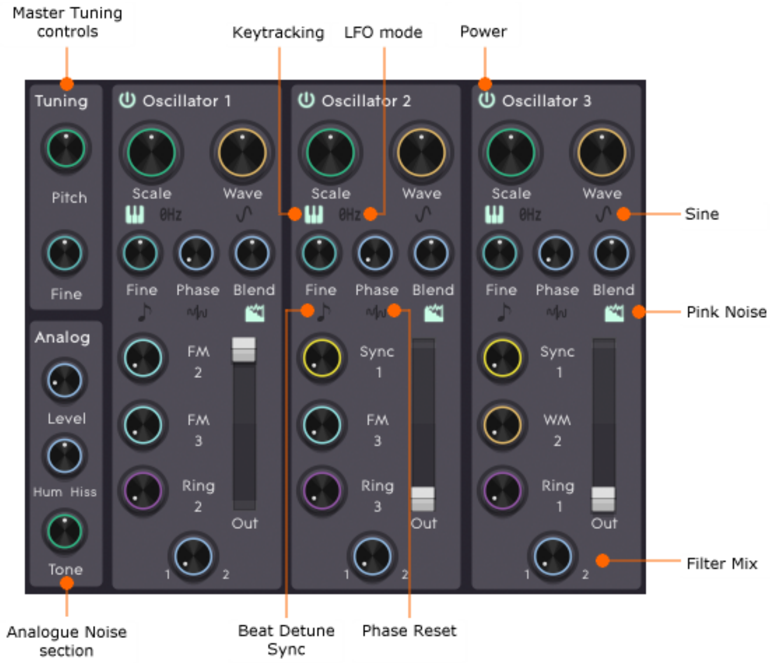
The Amp Envelope is always directly routed to Cypher2's **Amp** parameter with 100% depth. It can also be routed to other parameters in Cypher2 via the TransMod system.

Scope panel

As well as the Visualizer display, the Scope panel also provides several additional modulator editors - the Euclid, KeyZone, Maths and Curve panels provide additional modulation processing functionality.

3.1 Oscillator and Analogue Noise sections

Overview



Tuning

This section can be regarded as the 'master' tuning section.

Oscillator 1 / Oscillator 2 / Oscillator 3

These sections provide a variety of controls for each of the 3 oscs

Analog (Analogue Noise)

This section provides the Analogue Noise controls, which should be considered differently to noise as an oscillator. The latter is set using the **Blend** control for each oscillator.

Tuning and Oscillator sections

Pitch, Fine (master Tuning section) Scale (relative tuning for each oscillator)

The tuning of each of the 3 oscillators is dictated with their **Scale** controls which act relative to the master tuning (set using the **Pitch** and **Fine** controls).

The Pitch and Scale controls can be set in semitones or in just harmonics with optional snapping to whole semitones/harmonics depending on the current Unit/Snapping mode.

By setting the Harmonic mode on a pitch control's parameter context menu, each osc's **Scale** control can operate in harmonic pitch ratios instead of semitones, useful when dealing with FM and other audio-rate modulation processes.

Modulating the **Fine** control by small amounts with random TransMod modulation sources is useful for creating effects reminiscent of oscillators with unstable tuning.

Power ⏻

Each osc features a **Power** ⏻ button to activate it. With the button turned off, the osc is deactivated.

When deactivated, an osc cannot have any effect on inter-osc modulation such as FM, WM and Sync. In order to use an osc for these modulation processes but exclude it from the audio mix, activate its **Power** button but fully reduce its **Out** parameter so that it is not heard in the voice directly.

Wave

The shape of each oscillator can be varied continuously between Triangle or Sine (see below), Saw, Square and Pulse waveforms, using the **Wave** control.

Due to their continuously variable nature, Cypher2's oscs do not feature a dedicated pulse width control – in order to vary pulse width, the Wave parameter must be modulated within the square/pulse region (upper half) of its travel.

The waveform of Osc3 can be varied at audio rate by Osc2 using the **WM 2** control.

Sine

With an osc's **Sine** button activated, it is switched to sine core mode - the minimum position of the **Wave** control results in the osc producing a sine waveform. With the button deactivated, a triangle waveform is generated instead.

FM 3 / FM 2 / WM 2

Audio-rate modulation refers to modulating a parameter at the very fast speeds of audio waveforms, as opposed to the slower speeds of LFOs and envelopes (known as control-rate modulation). The audio-rate modulation in Cypher2 is the result of extremely detailed modelling, and strives to reproduce the complex behaviour that results when modulating analogue components at audio-rate.

Here's a summary of what Cypher2 can do in this respect:

- Audio-rate frequency modulation of Osc1 by Osc2 using the **FM 2** control on Osc1
- Audio-rate frequency modulation of Osc1 by Osc3 using the **FM 3** control on Osc1
- Audio-rate frequency modulation of Osc2 by Osc3 using the **FM 3** control on Osc2
- Audio-rate waveform modulation of Osc3 by Osc2 using the **WM 2** control on Osc3

Sync 1

The **Sync 1** controls in the Osc2 and Osc3 sections allow the frequency of these oscs to be synchronized to Osc1's frequency. Rather than a simple on/off button for this function, Cypher2 features a continuous control that provides full 'hard' sync at 100% and variable degrees of sync between 0 and 100%. These variable amounts of sync can provide interesting 'soft sync' textures and harmonic effects.

Fine / Beat Detune Sync ♪

The **Fine** control allows one oscillator to be detuned against another while keeping the rate of the resulting 'beating' constant across the keyboard. This opens up new rhythmic ways of playing sounds with classic detuned-beating characteristics.

This control also sets the osc's frequency (or rate) while the Osc is in LFO mode (when the **Low Frequency** button is enabled).

With the **Beat Detune Sync** ♪ button activated, the beating rate is set in BPM units with the **Fine** control, allowing tempo-synced effects.

Note that regardless of whether the **Fine** control is used, the beating rates between oscs are available as TransMod ramp-wave modulation sources.

Phase / Reset

Each osc's phase can be set with the **Phase** parameter.

With the **Reset** button activated, the phase is reset to the value defined by the Phase parameter on each envelope retrigger event (which is dependent on the **Retrig** button).

Disabling the Reset button means that the phase is effectively free-running.

The Phase control can be continuously modulated in real time for phase modulation effects (at control-rate rather than audio-rate).

Low Frequency 0Hz

Activating the **Low Frequency** 0Hz button on an osc switches it to LFO mode, meaning that the osc operates at lower-frequency control rates. With this mode activated, the **Fine** parameter controls its rate in Hz. Each osc is available as a control-rate TransMod source for use with this function, although the hard-wired modulation such as FM, WM and Sync can also be used, with the obvious caveat that these operate at control rates (non-audio rates).

Key

Activating an osc's **Key** button enables keytracking (modulation of pitch by the keyboard). With this button deactivated, the osc's pitch is not affected by the pitch of incoming MIDI notes.

Blend / Pink Noise

The **Blend** control crossfades the final audio output from each oscillator between:

- another osc sampled and held at audio-rate by the osc's frequency (fully left)
- the oscillator itself (centre position)
- white noise (fully right) or pink noise if the **Pink Noise** button is activated

Audio-rate sample & hold

The audio-rate sample & hold (S+H) function at the hard-left position of the **Blend** control produces the following signals:

Osc1	Osc2 sampled & held by Osc1's frequency. Osc1 is the clock trigger acting on Osc2
Osc2	Osc3 sampled & held by Osc2's frequency. Osc2 is the clock trigger acting on Osc3
Osc3	Osc1 sampled & held by Osc3's frequency. Osc3 is the clock trigger acting on Osc1

The S+H signal is a type of audio-rate modulation which is always active unlike, say, FM modulation – the amount of which is adjusted via the **FM 2** or **FM 3** controls on oscs 1 and 2. It uses the frequency of one oscillator as a clock signal to snapshot the amplitude of another oscillator's waveform output on each clock pulse. This value is then held until the next clock pulse is received, when the process repeats.

Because the modulating oscillator provides the frequency of the clock signal for the S+H function it has a very prominent effect on the pitch of the resulting signal. It is therefore presented within the modulator osc's **Blend** control, which allows a mix between this signal and the oscillator itself within the lower 50% of the Blend control's travel.

Noise: White and Pink

The upper 50% of the Blend control's travel allows a mix between the oscillator and a noise signal. By default this is a white noise signal. If the **Pink Noise** button is activated, a pink noise source is provided instead. Pink noise possesses equal energy across the frequency range, effectively meaning that more low-frequency content is apparent compared to white noise, which is more akin to a typical 'hiss' sound.

Ring (Ring modulation)

Each oscillator features a ring modulation function, allowing it to be multiplied by one of the other oscillators.

The ring-modulated signal is a result of bipolar multiplication of the output from 2 oscillators - the level of this signal within each osc's output is set using the **Ring** control. The ring modulation process and **Ring** level control occurs after the **Blend** control in the signal path. Each osc's **Ring** control sets the level for the ring-modulated signal from the following:

Ring 2 (Osc1)	Osc2 x Osc1
Ring 3 (Osc2)	Osc3 x Osc2
Ring 1 (Osc3)	Osc1 x Osc3

If the oscs' frequencies are harmonically related, the result is a 'musical' sound which can often resemble chords; if not, the output may sound dissonant and 'clangorous'.

Out

The **Out** control sets the final output level of each oscillator.

At a setting of zero, the osc is not heard in the final output. However, its effect is still heard if it is modulating one of the other oscs using the FM, WM, Sync or Ring functions.

To completely remove the effect of an oscillator in the voice, deactivate its **Power** ⏻ button.

Filter Mix

Each Osc's **Filter Mix** control determines the amount of signal from the osc that is routed to each of the two filter/shaper paths. When at the centre position, the osc's signal is sent equally to both filter/shaper paths.

Analog (Analogue Noise) section

The Analogue Noise section simulates the effect of noise and mains hum in certain parts of the audio and control signal paths, something that usually occurs in real analogue synth circuits to some extent.

Level

This control dictates the level of the Analogue Noise section's signal leaking into the audio signal path.

Hum Hiss

This control sets the degree to which the analog noise affects Cypher2's audio and control circuits. At lower settings, it leads to a subtly gritty and slurring character, while higher settings result in a more unstable and noisy sound.

Tone

The **Tone** control applies a 1-pole low-pass filter on the analog noise signal for dialling out higher frequencies and harmonics as required.

3.2 Shaper, Filter and Output sections

Overview

Shaper and Filter sections

Cypher2's dual Shaper/Filter sections can be considered as the 'tone-shaping' part of the synth voice, within which the harmonically complex output from the oscillators is sculpted by circuit-modelled tone modifiers.

The input to each of the Shaper/Filter paths is dictated by the Oscillator section's **Filter Mix** controls.

Shp 1 & Filter 1 sections

These sections control the Shaper 1/Filter 1 path. Shaper 1 is placed before Filter 1 unless Shaper 1's **Post** button is activated, in which case it is placed after Filter 1.

Shp 2 & Filter 2 sections

These sections control the Shaper 2/Filter 2 path. Shaper 2 is placed before Filter 2 unless Shaper 2's **Post** button is activated, in which case it is placed after Filter 2.

VCF 1+2 section

This section provides a way of adjusting (and modulating with Envelopes 1 and 2) the cutoff of both Filter sections simultaneously. The Filter section's **Link** button must be activated for it to be affected by this section.

Output Mixer-Amplifier section

The Output section provides control over Cypher2's 'quad-stereo' final V.C.A., which allows routing signals from 4 separate points in the voice circuit into the Effects section - before and after Filter 1 and Filter 2.

The final V.C.A. can be considered as the part of a synth which 'articulates' the voice - it 'opens up' the signal, which is always running underneath, like opening a door to allow something to pass through it.

Shaper sections (Shp 1 / Shp 2)

A waveshaper applies a mathematical function to the incoming waveform in order to alter its harmonic content and introduce distortion. It is useful for overdriving the signal to add abrasive grit and character. Some of the waveshaper types are 'wavefolder' circuits, which are typically applied to harmonically-simple waveforms (such as sine / triangle shapes) to create additional harmonics.

Shaper Power

To enable each of the waveshaper blocks, engage its **Power** button.

Post

With its **Post** button deactivated, the shaper block is placed before the corresponding filter block (for example, Shaper 1 > Filter 1). By engaging the **Post** button, the shaper is moved after the filter (Filter 1 > Shaper 1).

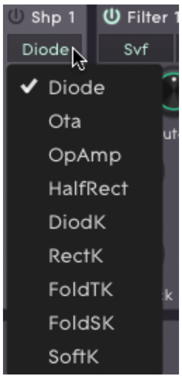
Mode

This control selects from a number of distinct waveshaper and wavefolder models, each providing its own flavour of distortion and added harmonic content.

The *Diode*, *OTA*, *OpAmp*, *HalfRect*, *DiodK* and *RectK* modes each provide a different type of overdrive/ distortion, the characteristics of which are dictated by the reaction of the waveshaper's modelled components.

The *FoldTK* and *FoldSK* modes are classic 'wavefolder' type functions. These are rarely found in pre-patched synthesizers and are most commonly associated with modular synth systems - examples include the Serge Wave Multiplier and close derivatives in Eurorack and other systems.

The *SoftK* mode is a soft-saturating clipper circuit.



Diode	Classic diode-based clipper model with fixed filter
OTA	Operational Transconductance Amplifier chip model
OpAmp	Operational Amplifier chip model
HalfRect	Half-rectifier circuit model
DiodK	Diode clipper with keytracking filter*
RectK	Rectifier with keytracking filter*
FoldTK	Triangle-based wavefolder with keytracking filter*
FoldSK	Sine-based wavefolder with keytracking filter*
SoftK	Soft-saturating clipper with keytracking filter*

* shaper Types with the K suffix all feature a keytracking 1-pole tilt-style EQ-shelf filter adjusted using the **Tone** control, tuned to 2 octaves above the master Tuning setting. All other types feature a fixed filter accessed with the **Tone** control.

Drive

The Drive control increases the gain of the signal going into the waveshaper – higher values drive the waveshaper harder, leading to more distortion and added harmonics.

Tone

For the *Diode*, *OTA*, *OpAmp* and *HalfRect* **Mode** settings, the **Tone** parameter applies a 1-pole Low-pass filter after the waveshaper for dialling out higher frequencies and harmonics as required.

For the *DiodK*, *RectK*, *FoldTK*, *FoldSK* and *SoftK* **Mode** settings, the **Tone** parameter is switched to control a special keytracking 'tilt'- style shelf-EQ filter, which effectively allows the spectral emphasis to be adjusted after the waveshaping is applied to the signal.

The lower part of the control emphasizes lower frequencies and de-emphasizes higher frequencies, with the upper part doing the opposite. Since this filter tracks the keyboard, its effect is always consistent across the keyboard range.

Filter sections (Filter 1 / Filter 2 / VCF 1+2)

Filter Power ⏻

The **Power** ⏻ button activates or deactivates the filter. When deactivated, incoming audio passes through the filter stage unaffected.

Cutoff / VCF1+2 / Link ⇉ ⇇

Filter 1 and Filter 2 both possess a **Cutoff** control which adjusts its individual cutoff frequency.

Each filter's cutoff frequency is also affected by the **VCF1+2** control if its **Link** button is activated. In this scenario, each filter's **Cutoff** control offsets its cutoff frequency relative to the **VCF1+2** control by positive or negative amounts. At the centre position, the filter's cutoff is the same as that of the **VCF1+2** control.

With its **Link** button deactivated, the filter's cutoff is not affected by the **VCF1+2** control.

Env1 & Env2

Envelopes 1 and 2 are directly routed to modulate the **VCF1+2** parameter (the cutoff frequencies of both filters simultaneously).

The **Env1** and **Env2** controls adjust the depth of modulation from Envelope 1 and Envelope 2 respectively.

Route

The routing of the filters can be changed between 2 serial and 2 parallel configurations using the **Route** parameter.

This parameter is actually located within the Output Mixer-Amplifier section.

1>2	Serial, Filter1 before Filter 2
2>1	Serial, Filter 2 before Filter 1
1+2	Parallel: Filter 1 + Filter 2
1-2	Parallel, Filter 1 + inverted Filter 2

Width

The **Width** control (like the **Route** control, this is located within the Output Mixer-Amplifier section) sets the amount of stereo spread between the filters when using either of the parallel **Route** settings (*1+2* or *1-2*).

Reso (Resonance)

The **Reso** control adjusts the resonance of each filter. Higher settings result in self-oscillation.

Drive

The **Drive** control increases the gain of the signal entering the filter. This leads to the filter circuit being overloaded, drastically altering its tone and character. High Drive amounts can limit the apparent amount of resonance in the filter – for warmer resonant sounds, turn down the Drive control, making up the volume if necessary with the **Amp** or **Level** controls in the Output section.

Keytrack

The **KeyTrack** parameter controls the amount of direct modulation from keyboard pitch, which is applied relative to the filter's Scale setting.

FM 3

The cutoff frequency of each filter can be modulated at audio-rate by the frequency of oscillator 3. The **FM 3** controls adjust the amount of frequency modulation for each filter.

Type

Select from various classic filter designs for either of the two filter paths using the **Type** parameter:

Svf	Classic 'state-variable' filter design, derived from an American synth module with 2-pole multimode filter
Mgf	'Ladder' filter model based on a famous American monosynth
MgX	Alternative ladder-filter model with a more aggressive clipper in the feedback circuit, giving brighter resonance / harder edge to the sound
Fat	'Sallen-key' filter model derived from a classic Japanese semi-modular monosynth
Jpt	Cascaded OTA filter model based on a vintage Japanese polysynth filter chip
Comb	Comb filter model with a number of modes, useful for varied timbres and karplus-strong synthesis

Mode

The **Mode** control allows any of the following filter modes to be selected for the *Svf*, *Mgf*, *MgX*, *Fat* and *Jpt* filter Types:

L2, L4	2-pole (12 dB/oct.) & 4-pole (24 dB/oct.) Low-pass filters
B2, B4	2-pole & 4-pole Band-pass filters
H2, H4	2-pole & 4-pole High-pass filters
P4	4-pole Peak filter
N4	4-pole Notch filter

The *Comb* filter Type provides the following Mode settings:

+Ve	Positive feedback loop
+VeD	Positive feedback loop with internal damping
+VeL	Positive feedback loop with low-pass filter in series
+VeH	Positive feedback loop with high-pass filter in series
-Ve	Negative feedback loop
-VeD	Negative feedback loop with internal damping
-VeL	Negative feedback loop with low-pass filter in series
-VeH	Negative feedback loop with high-pass filter in series

Output Mixer-Amplifier section

The Output Mixer-Amplifier section (or simply the Output section) represents the final output of the synth circuit before it enters the FX processing blocks. It performs the function of a synth's output V.C.A. - a 'voltage controlled amplifier'. A synth's output V.C.A. reacts to a voltage - a control signal typically from an envelope - to control the amplitude of the synth voice. It effectively articulates the loudness of the synth voice - the manner in which it begins (from a previous state of silence) and continues over time until it ends, again with a state of silence.

It may be helpful to consider that a hardware VCA is often simply a voltage controlled attenuator which is fully closed - set to the minimum value - by default, meaning that it fully attenuates the signal so that it is muted. When a control signal is received from an envelope, the attenuator is opened accordingly so that the synth voice can be heard. Some V.C.A. circuits may simply go up to a maximum of unity gain, while others can introduce additional gain to distort their circuitry and produce additional harmonics that may contribute to the quality of the overall timbre.

Please note that Cypher2's VCA is hard-wired to the output of the Amp Envelope, meaning that the Amp Envelope always controls the articulation of each voice.

While Cypher2's V.C.A. has some significant 'bells and whistles', it ultimately performs the same function as any synth's final V.C.A.

It provides several output 'taps' from different stages of the voice circuit, each with a variable output level.

• **Filter1 Pre**: Signal entering the Filter1 waveshaper / filter path

• **Filter1 Post**: Signal after processing by the Filter1 waveshaper / filter path

• **Filter2 Pre**: Signal entering the Filter2 waveshaper / filter path

• **Filter2 Post**: Signal after processing by the Filter2 waveshaper / filter path

These signals are routed to the A and/or B effects chains on a per-voice basis, according to the current setting of each Shaper/Filter path's **A-B Route** control.

It is important to remember that the overall articulation of the voice is still controlled by the main **Amp** and **Level** controls. However, the 4 individual controls for each tap (which can also be modulated) act as relative offsets to the effect of the **Amp** and **Level** parameters.

The 4 output taps are *stereo* with the position of the signal dictated with the **Pan** control. Cypher2's V.C.A. can therefore effectively be considered as a 'quad stereo VCA'.

The **Amp** circuit -itself features modelled analogue non-linearities in the form of a waveshaping stage between the Amp and final **Level** controls. This part of the circuit saturates and distorts with higher input levels, meaning that it can be overloaded like a classic analogue synth V.C.A.

• For a cleaner sound, keep the **Amp** control at low settings and increase the **Level**.

• To overload the amp, turn up the **Amp** parameter and turn down the **Level** control as required to prevent clipping.

Amp

The **Amp** parameter is directly modulated by the Amp Envelope. Its value represents the amplitude at the maximum value of the Amp Envelope (at the end of its Attack stage) and hence controls the loudness of each voice.

Although the Output section features 4 separate level controls which 'tap' the signal from various parts in the voice's signal path (before/after both Shaper/Filter paths), the **Amp** parameter controls the overall level of the voice which is heard within these 4 taps.

Pan

The **Pan** parameter sets the position of the voice in the stereo field.

Try modulating this control with the TransMod system using a Voice or Unison modulation source – this leads to a rich stereo spread of chords or multiple unison voices.

Level

This parameter sets the final **Level** of each voice before it is summed with all other active voices. If this parameter's value is set too high, the output of Cypher2 entering the Effects section may clip.

Unlike the **Amp** parameter, the **Level** stage is a clean digital gain control – it does not feature any saturating characteristics when overdriven. It is intended to be used to compensate for the Amp parameter setting as described above.

Suitable settings for this parameter depend upon context - care should be taken especially with polyphonic sounds to allow enough headroom for the output not to clip when multiple voices are mixed.

Filter1 Pre

This control routes a variable level of the pre-Filter 1 signal – in other words, the signal entering the Shaper 1 / Filter 1 path – to the effect section. The destination of this signal depends on the state of the **Filter1 A-B Route** control.

Filter1 Post

This control routes a variable level of the post-Filter1 signal – in other words, the output from the Shaper 1 / Filter 1 path – to the effect section. The destination of this signal depends on the state of the **Filter1 A-B Route** control.

Filter1 A-B Route

This control dictates whether the Pre and Post outputs from the Shaper 1 / Filter 1 path are routed to Effects Bus A (minimum position), Effects Bus B (maximum position) or a variable proportion between the two (positions between min and max).

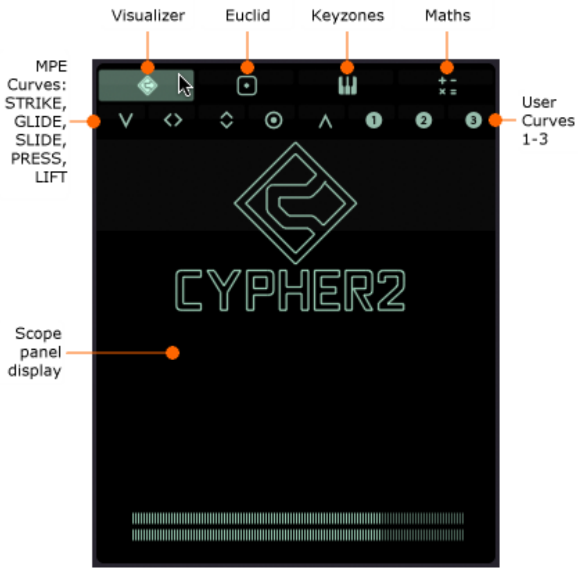
Filter2 Pre / Filter2 Post / Filter2 A-B Route

These parameters provide the same functions for the Shaper 2 / Filter 2 path as the corresponding **Filter1 Pre**, **Filter1 Post** and **Filter1 A-B Route** controls for the Shaper 1 / Filter 1 path.

3.3 Scope / Visualizer panel

The Scope panel performs a number of functions in Cypher2, with the panel display changing depending on which of the buttons at the top of the panel is selected.

Visualizer



By default the Visualizer panel is shown (with the **Visualizer** button activated as shown on the left). This mode performs 2 functions:

1. By default it displays the Cypher2 logo graphic, a level meter representing the final audio output and the Description for the current preset (if it was saved with one).

2. When the cursor is moved above or used to adjust certain controls, a number of graphic visualizer displays are provided.

The Visualizer represents various parts of the Cypher2 synthesis engine. Note that it should not be regarded as an 'oscilloscope' but rather as an aid to visualising each part of the synth and current signal routings.

Modulation processors

As well as the Visualizer, the Scope panel provides several additional modulator editors:

Euclid processor

This processor is a combination of an X-Y pad with additional modulation processing with inertia and slew.

KeyZone processors

These editors allow custom keytracking responses to be defined.

Maths

This editor allows complex modulation processing with mathematical functions for multiple sources.

Curve processors

These editors provide the ability to remap and slew the shape of a modulation source.

Visualizer in detail

The Visualizer display is updated in real time as any relevant controls are adjusted and also shows the effect of any modulation routed to relevant parameters. It is divided into multiple sections:

Upper section

This section shows context-sensitive displays such as the final output wavefom for oscillators and LFOs, envelope shapes, filter responses and so on. The display shows the last section over which the cursor is moved, unless the **Lock Scope** function (located in the Parameter context menu) is used for any control.

Lower section

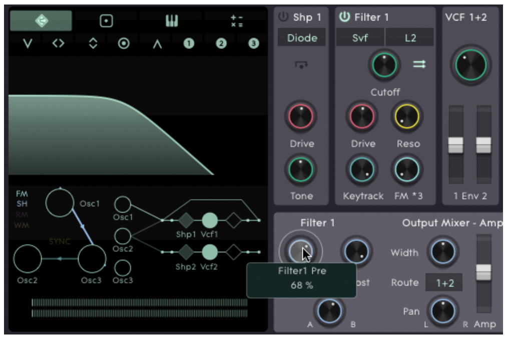
This section can show either of the following depending on the area under the cursor:

• the various inter-oscillator modulation routings on the left and the synth voice's audio path on the right

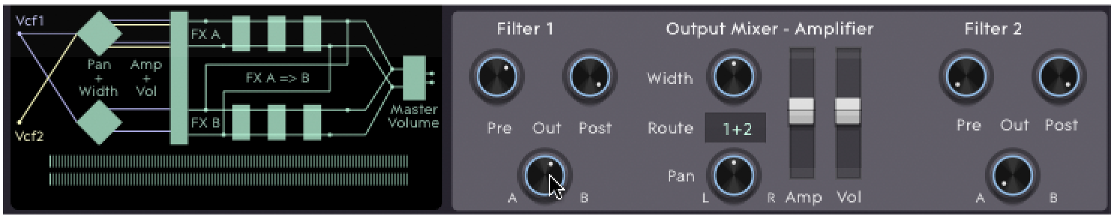
• the output/effects audio paths

• **upper section**: shows Osc3 output waveform

• **lower section**: shows inter-osc modulation and audio path routings (the synth voice in this case)

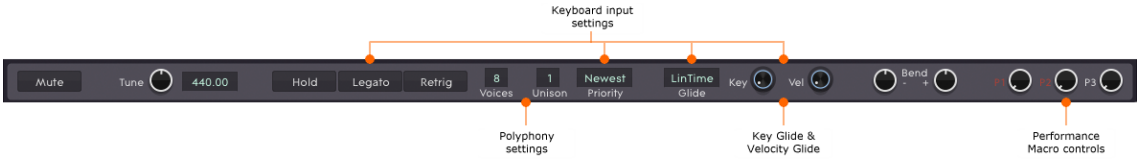


• this screenshot shows the Pre-Filter 1 routing appearing on the synth voice audio path visualizer as the control is moved



• this screenshot shows the output/effects audio path routing when the Output section's **A-B Route** control is used.

4 Global Synth Controls



This section of the interface is always visible, even when viewing the Sequencer or Effects pages. It is used for various global synthesis settings and controls.

Polyphony settings

The **Voices** and **Unison** settings are covered in the next section.

Keyboard input settings

The **Priority**, **Gliding**, **Key Glide**, **Velocity Glide**, **Hold**, **Legato** and **Retrig** controls are involved in adjusting various aspects of how Cypher2 responds to keyboard input. These are covered later in the chapter.

Mute

The **Mute** button shuts down any playing notes/voices - it can be considered as a 'panic' button.

Tune (Master Tuning)

This control sets the master tuning - it specifies the frequency for the A note above middle C. The default is 440 Hz and is adjustable between 420 and 460 Hz.

Bend Up
Bend Dn (Down)

These controls set the pitch-bend sensitivity – the amount of pitch modulation from MIDI pitch-bend messages – on MIDI channel 1. This channel is intended for using pitch-bend with conventional '2D' MIDI controllers. The amount of upward and downward pitch displacement can be specified separately, to a maximum of 12 semitones in each direction.

Other MIDI channels (2-16) provide MPE-based polyphonic pitch-bend messages and are not affected by the **Bend Up** and **Bend Dn** controls. Pitch-bend for these channels is handled through the *PlBend+* source and GLIDE Curve processor (accessed with the *PlGlideCurve* source). With default ROLI Dashboard settings, the PlGlideCurve source should be set to modulate the master Tuning section's **Pitch** control by 48 semitones.

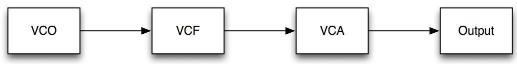
P1 (Perf1), P2 (Perf2) & P3 (Perf3)

These controls represent the 3 available performance macro controllers

To map these controls to the hardware controllers available to you, assign them using the MIDI Learn function.

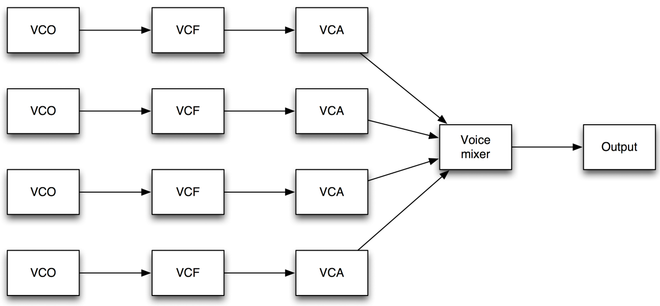
4.1 Polyphony: Voice and Unison settings

Polyphony overview



A synth voice can be considered to be an entire monophonic synth that plays a single note at a time – a monophonic synth features 1 voice.

• Monosynth architecture



• Polysynth architecture

A polyphonic synth features multiple voices, meaning that it is comprised of multiple identical monosynths with a simple logic circuit for distributing received notes amongst them.

The most obvious example of the latter is the original family of Oberheim 2-voice / 4-voice / 8-voice polyphonic synths. These were simply multiple discrete Oberheim SEM units housed together with note-distribution and some other global programming functions.

While the original Oberheim polysynths are perhaps the most obvious examples for visualizing the architecture of a polyphonic synth, this architecture is common to all conventional polyphonic analogue synths. Usually it is implemented in the form of 'voice cards' which contain the circuitry for each individual voice along with a single control panel which distributes parameter edits to the voice cards.

The same principles are used within Cypher2.

Voices & Unison controls

These numerical text-boxes set the number of active **Voices**.

With Unison set to 1, the **Voices** setting represents the maximum number of simultaneous notes (polyphony) that can be played.

Voices for the current patch and the number of **Unison** voices if required.

If more simultaneous notes are played than the current voice and unison settings allow, voice-stealing functions are applied, the behaviour of which is specified by the **Priority** control (see the next section).

However, when using a **Unison** setting higher than 1, the polyphony is the number of Voices divided by the Unison setting.

Unison

The number of **Unison** voices is a sub-set of the maximum number of voices - it determines how many available voices to stack for each note. For example:

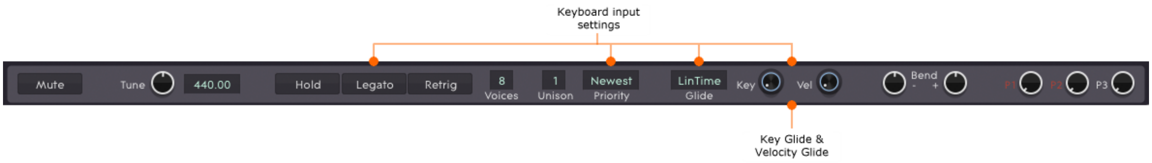
• specifying 8 voices and 2 unison voices results in 4-note polyphony, with each note comprising 2 stacked unison voices

• specifying 12 voices and 3 unison voices also results in 4-note polyphony, with each note comprising 3 stacked unison voices

Voices, Unison and TransMod modulation

In traditional analogue synths, unison voices could usually only be detuned against each other. The possibilities in Cypher2 are far more advanced - using the polyphonic Unison TransMod sources in Cypher2, any synth parameter - including but not limited to oscillator pitch - can be spread across a range for each unison voice. Similarly, a number of Voice TransMod sources are also provided in order to create per-voice modulation with regular polyphonic playing (1 voice per note).

4.2 Keyboard Input, Glide and Retrigger controls



Keyboard input settings

Priority

The **Priority** control dictates the voice stealing priority if more notes are played than the number of voices available and can be set to *Newest*, *Oldest*, *Highest*, *Lowest*, *Hardest* or *Softest*.

The default setting is *Newest*, which means that new notes are prioritized over old notes. If **Voices** is set to 1 (see the last section for details of this setting), playing a note while another note (the older note) is already playing results in the new note 'stealing' the voice as it has a higher priority. When this new note is released, the older note is played again.

The *Lowest* setting results in the lowest played note always being given priority - this is similar behaviour to that in the original Minimoog. The *Hardest* and *Softest* settings give priority to higher and lower velocity notes respectively.

Hold

Engaging the **Hold** button results in incoming MIDI note *off* messages being ignored until the button is deactivated. Therefore, if the Hold button is enabled, any keys already playing or subsequently played are sustained indefinitely, even after the keys are released.

Please remember that held notes are not saved with presets!

Glide and Retrigger settings

Classic analogue synth glide essentially smooths transitions between keyboard-input note intervals, leading to the typical 'portamento'-style rising or falling pitch effects.

Cypher2 can additionally apply Glide to Velocity TransMod modulation. When activated, any modulation which is derived from the *OnVel+* source (and any other source derived from this, such as the MPE STRIKE curve) is smoothed over time with new incoming events.

Note that glide behaviour is also affected by the **Legato** button (see below).

Key (Pitch Glide)

The **Key Glide** control sets the glide time towards the pitch of new note events.

The control can be modulated with the TransMod system - try using a VRand source to randomize the glide time on each note!

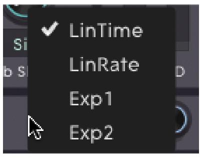
Vel (Velocity Glide)

The **Velocity Glide** control sets the glide time towards new MIDI note velocity (OnVel+ TransMod source) modulation depths on each key-on event.

Since note velocity is perhaps the most immediate performance control method, Velocity Glide offers an easy way of injecting additional variation and dynamic response to a synth performance. It is also polyphonic so occurs independently on each played voice.

For example, if the OnVel+ TransMod source is set to modulate the filter's **Cutoff** parameter so that it increases with higher velocity, playing a harder note after a soft note would cause the cutoff frequency to glide to the higher value over the **Velocity Glide** time period. If the Velocity Glide control is set to zero, the cutoff frequency is fully modulated by a new note as soon as it is played.

Like the **Key Glide** control, the **Velocity Glide** control can itself be modulated by the TransMod modulation system. Again, a VRand source can be very effective, or try a Unison source in a unison-enabled patch so that each playing voice features varying glide times.



Glide (Glide Mode)

The **Glide** control changes the glide time response between Linear (*LinTime*, *LinRate*) and Exponential (*Exp1*, *Exp2*) settings. Each setting results in a different shape and resulting playing feel for the 'curve' of glide transitions.

Legato
Retrig

These buttons provide further options for Glide and modulator retriggering.

The **Legato** button relates to glide: activating it results in a 'fingered' glide – meaning that glides only occur when 2 notes 'overlap'. This applies to both Velocity Glide and Pitch Glide.

The **Retrig** button forces gated modulators (Ramp/LFO/ModEnv/AmpEnv) to retrigger from their current value when voice stealing occurs. With envelopes, the Attack stages of the retriggered envelopes are started *from the stolen voice's current envelope positions*. This means that if any envelope has not completed its Attack stage, the retriggering will effectively not be explicitly heard - the previous envelope effectively continues.

4.3 MIDI Learn mode

Cypher2 allows parameters to be mapped to MIDI CCs for remote editing and performance. In particular, the **P1**, **P2** and **P3** macro controls and the **Base X** / **Base Y** controls in the Euclid processor are intended to be mapped to MIDI CCs of your choice.

2D and 5D MIDI CC Maps

Cypher2 provides 2 distinct CC Learn configurations, called CC Maps, intended for use with regular MIDI keyboards (2D) and MPE controllers (5D). This is to ensure that MPE controller MIDI messages which are used for 'voice-per-channel' polyphonic MPE operation do not cause playability problems when used with 2D sounds which may be intended to be played using the same controller messages mapped to the *monophonic* macro controls.

The current MIDI Learn CC Map is specified using the **2D**, **5D** and **Auto** buttons in the upper part of the Browser.

If the **Auto** setting is selected, the MIDI Learn setup is changed dynamically on each preset load, depending on whether the preset name ends with ' 5D', with the status at any given time indicated using an 'underscore' symbol underneath the 2D or 5D buttons.

Note that the keyboard in the lower part of the Browser / Easy page changes between a 'piano'-style keyboard view and a Seaboard-type MPE keyboard view depending on the currently selected MIDI mapping setup.

Creating MIDI CC assignments

1

First, ensure that the required MIDI Learn setup is selected by clicking either the **2D** or **5D** button in the upper part of the Browser. The **Auto** mode is not suitable for creating assignments.

2

Click the **MIDI Learn** button to enter MIDI Learn mode. Parameters in Cypher2 which can be mapped to MIDI CCs are highlighted in green.

3

Click the parameter to map on the interface.

4

Move the desired physical MIDI CC knob, slider or other controller.

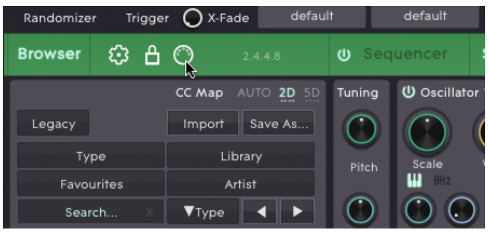
5

The parameter is now mapped to the controller. The MIDI CC number is overlaid on the control.

6

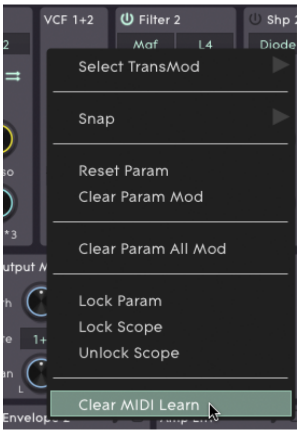
Alternatively, double-click the desired control to type a MIDI CC number manually, then press ENTER.

Double-clicking to manually enter a CC may be useful with joyticks/X-Y pads which may involve difficulty in outputting a single axis for the Learn function.



7

Click the **MIDI Learn** button again to exit MIDI Learn mode.



Removing MIDI CC assignments

To remove MIDI CC Learn assignments, right-click on the desired control to display the Parameter context menu and use the **Clear MIDI Learn** function.

This operation can be performed whether MIDI CC Learn mode is currently activated or deactivated.

5 Modulation

Modulation can be considered as the movement of a parameter over time. Cypher2 features 2 ways of modulating parameters:

- Direct modulation of selected parameters for basic functions
- TransMod modulation, which offers a hugely increased array of modulation possibilities

1. Direct modulation for selected parameters

Cypher2 features direct modulation routings for selected destination parameters - Oscillator tunings, the cutoff frequency of both filters and the voice amplitude represented by the Output Mixer-Amplifier section's **Amp** parameter.

Keytracking

This source tracks the keyboard (MIDI input note range) and features controls to enable/disable 100% modulation of oscillator pitch (with each osc's **KEY** button) and a variable amount of modulation for each filter's cutoff frequency with the **Keytrack** parameters.

For oscillators, activating the Key button means that MIDI note input produces chromatic pitches from the oscillator.

For filters, a 100% **Keytrack** setting means that the filtered output output remains harmonically constant throughout the MIDI note range and produces musical chromatic pitch when in a self-oscillating state (with the **Reso** control turned up to a high setting).

Mod Envelopes 1 & 2 to VCF1+2 (Cutoff control for both filters, used in conjunction with LINK buttons)

The Env1 and Env2 faders allow the VCF1+2 parameter to be modulated directly from Mod Envelopes 1 and 2 respectively. VCF1+2 affects both filters' Cutoff parameters if the respective LINK button is activated.

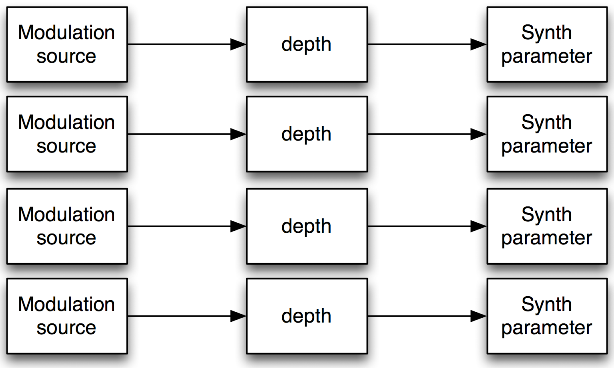
Amp Envelope to V.C.A. Amp parameter

This is a direct modulation routing which is always active with 100% depth - this means that the Amp Envelope always acts to open the VCA to allow Cypher2's output to be heard.

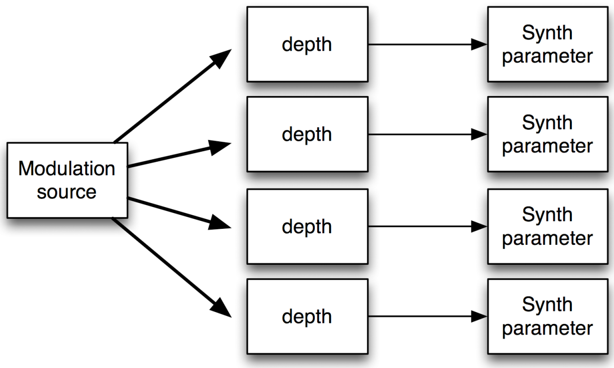
2. TransMod modulation

The TransMod modulation system must be used for all other modulation in Cypher2. It allows amazingly varied and complex modulation effects.

Essentially, TransMod is similar to the well-known 'mod-matrix' found in many synthesizers. However, instead of simply connecting a modulation source to a single synthesis parameter and setting the depth or amount of modulation, TransMod allows a single modulation source to be routed to multiple synth parameters with varying depths for each routing.



Traditional modulation matrix



Cypher2 TransMod system

Depths are shown visually directly on the interface itself when a source 'slot' is selected, rather than in an abstracted list.

The TransMod modulation system is covered in depth in the next section.

5.1 Using TransMod modulation

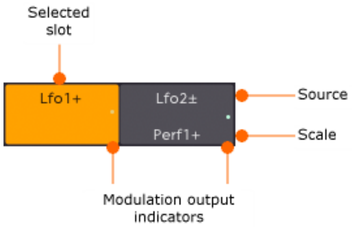
The TransMod modulation system centres around the 24 TransMod 'slots' in the upper part of the Cypher2 interface.

With no slot selected, adjusting Cypher2's parameters adjusts the 'initial state' of the synth - before any TransMod modulation is applied. Note that, as mentioned previously, Cypher2 contains some direct modulation routings that may already be active within this 'initial state'.

Slots 1-16 are freely-assignable slots - they are populated by a selection of sources in the default preset but can be reassigned to any of the wide range of available sources.

Slots 17-21 are fixed to MPE Curve processor sources. Slots 22-24 can be switched between the 3 User-defined Curve processor sources and the 3 Performance Macro sources in the Settings menu.

Using assignable TransMod slots



Slots 1-16 can each be assigned to any available TransMod modulation *Source* (specified in the upper part of the slot) as well as a multiplying *Scale* (in the lower part of the slot), also chosen from the available TransMod sources.

The far right of the slot features an LED-style indicator representing the output value from the slot. In this screenshot, the slot on the right features a bipolar LFO source - this is reflected in the modulation indicator which features 2 LEDs for the positive and negative portions of the bipolar output.

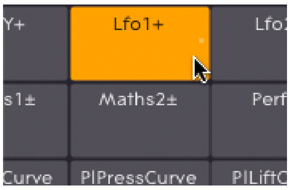
When a TransMod slot is selected Cypher2's parameters display any modulation 'amounts' that may exist from the slot to each control. The modulation amounts can be edited on the controls themselves using the outer ring for rotary controls or the slider path for vertical sliders.

With no TransMod slots selected, editing *only* occurs on the initial state of Cypher2's parameters.

Selecting a TransMod slot and assigning modulation sources

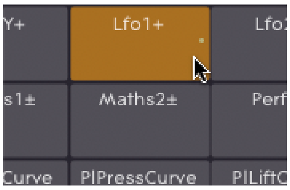
To begin using the TransMod system, a TransMod slot must first be selected.

Selecting a TransMod slot



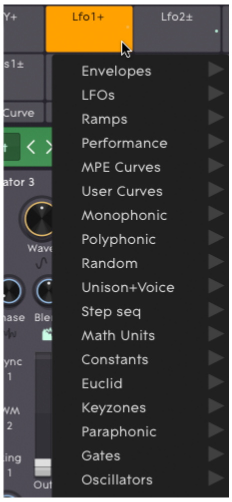
• to select a slot without displaying any menus, click anywhere within the slot.

The currently selected slot is highlighted in bright yellow.



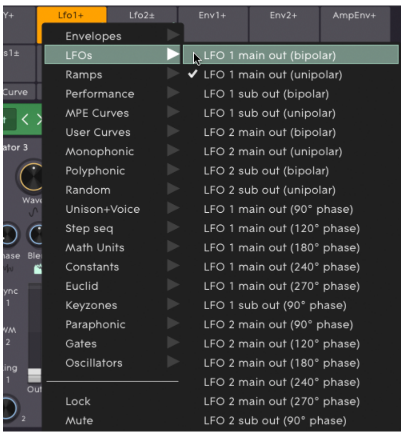
• to deselect the slot, click it again.

• **right-click** on the *upper* part of a slot to select it and display its TransMod *Source* menu.

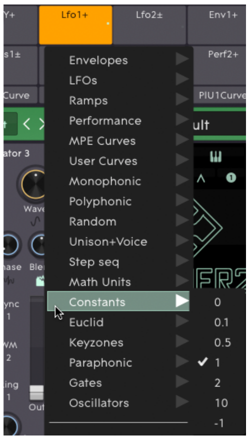


• **right-click** on the *lower* part of the slot to select it and display its TransMod *Scale* menu.

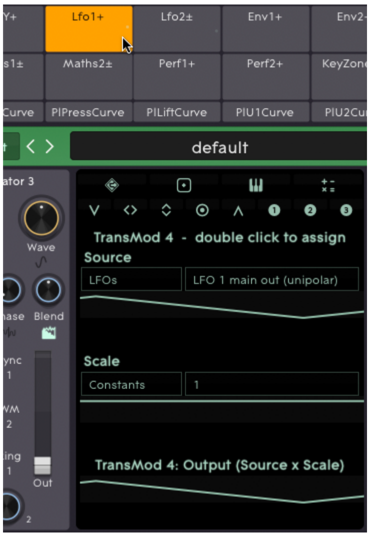
Assigning a modulation Source or Scale to a TransMod slot



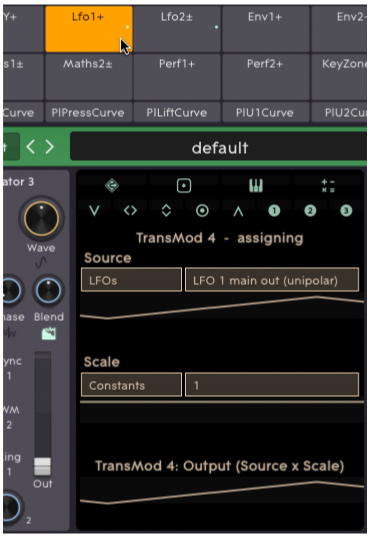
• When right-clicking to display the TransMod *Source* menu as described previously, simply navigate to the desired category sub-menu and click the required source.



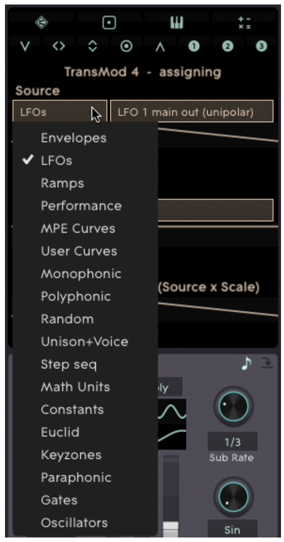
• Similarly, navigate to the desired *Scale* after right-clicking in the lower part of the slot. When nothing is displayed for the *Scale*, it is actually a Constant of *1* (meaning that the *Source* is unchanged by the *Scale*).



• Alternatively, when the cursor is moved over a TransMod slot, the slot's contents are shown in the Scope panel.



• Double-click to enter assign mode: the display in the scope changes colour and is locked to the Scope until the slot is deselected (or until another slot is selected).



• The category and source drop-down menus for the *Source* and *Scale* can now be used within the Scope panel.

TransMod slots can also be selected via the Parameter context menu or via the following keyboard shortcuts (in DAW/host applications which do not intercept keyboard input):

TAB or **]**	Next TransMod slot
[	Previous TransMod slot

Creating modulation amounts

Once a TransMod slot is selected, amounts of modulation can be created directly on Cypher2's parameters, as shown earlier in this manual.

These modulation amounts or 'depths' represent the greatest amount of movement of the parameter away from its initial value when the TransMod slot's modulator sends out a maximum value.

For 'continuous' modulation sources - such as envelopes, LFOs, and so on - the modulation depths represent the 'travel' of movement over time. While the modulation source varies over time, the values of any modulated parameters change proportionally within the defined range according to the values they receive from the modulation source.

For other modulation sources - such as Random, Voice and Unison sources - the modulation depths represent a range throughout which values are distributed or 'spread' at the initiation of each voice.

Compared to a traditional mod matrix, the TransMod system offers some advantages. It is easier to keep track of large numbers of modulated parameters. There is also no abstracted, table-based relationship between depths and parameters – all depths are shown visually, directly on the parameters themselves using the dual-action controls.

Almost all continuous synth parameters can be modulated using the TransMod system. The only exceptions are the following:

• buttons and any drop-down menu without a corresponding rotary control

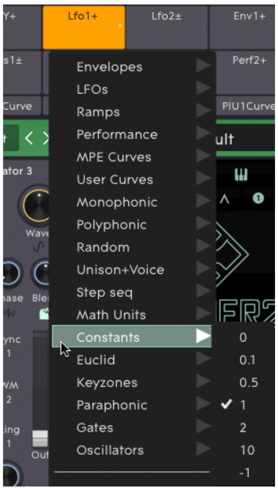
• **Gate source** controls for gated Modulators (LFOs, Envs etc)

• all parameters in the Global controls section except the **Vel Glide** and **Pitch Glide** controls

• Quick-preset and morphing controls

Scale functions

As previously mentioned, the lower part of each TransMod slot allows to specify a TransMod Scale function in the lower part of the slot, which multiplies the source specified in the upper part of the slot.



By default, this is set to 1 (a numeric constant) meaning that the modulation source is mapped 1:1 with any modulation amounts on destination parameters. In other words the source is unchanged by the scaling function.

Destination modulation depths can be multiplied by any available monophonic or polyphonic modulation source - an easy way to create more complex modulation behaviours.

Scaling examples

• Note-on velocity (*OnVel+*)

Scale any modulation source by the *OnVel+* source to dynamically affect the amount of modulation applied to each note with varying MIDI key velocity.

• Performance controller (*Perf1+*) or monophonic controller such as mod-wheel (*Mod+*)

Scale an LFO with the Mod+ source or a Performance controller mapped to the mod-wheel. This allows the mod-wheel to control the amount of LFO modulation.

• Use a Ramp (*Ramp+* or *Ramp2+*) to delay and fade in an LFO - see the example below

Simple TransMod examples

Using the TransMod system to create velocity-based filter modulation

1. Select *OnVel+* (Polyphonic menu) or *PlStrikeCurve* (MPE Curves menu) as a TransMod slot's source.

2. Turn down Filter 1's **Cutoff** parameter to a low value.

3. Draw an upwards modulation depth on the outer ring of the **Cutoff** control.

Now incoming MIDI note-on velocity (processed through the STRIKE Curve processor if using the *PlStrikeCurve* source) dictates the value of the **Cutoff** parameter for each played note.

Using the TransMod system to scale modulators

Using the Ramp to delay and 'fade in' an LFO

1. Deactivate the **Sync** and **Loop** buttons on the Ramp section if they are activated.

2. For Ramp 1, increase the **Delay** parameter to around *500ms* and increase the **Rise** parameter to around *1 second*.

3. Select the *LFO+* source in the upper part of a TransMod slot

4. Select *Ramp+* as a Scale function in the lower part of the same TransMod slot

5. Now set a TransMod modulation depth within this slot on a parameter such as the Filter **Cutoff** or Oscillator **Wave** control in order to hear the effect.

Managing TransMod modulation

Modulation indicators and slot/control highlighting

Cypher2 provides several ways of keeping track of controls that are being modulated from the various TransMod slots. The following example, with TransMod modulation amounts set up from 3 slots, demonstrates these functions:

• If any controls are modulated from the current slot, they are highlighted in orange in addition to the orange modulation indicators within the outer ring or slider path (beneath each modulated rotary control and above/below each slider). This is to aid visibility of very small amounts of modulation on controls.

• If a TransMod slot features any modulation depths, it is highlighted in *light grey*. This allows easy identification of which slots are currently modulating parameters.

In this example, only slots 4-7 feature any modulation depths to parameters.

Slot to parameter highlight

Hover the cursor over any other TransMod slot than that which is currently selected - any modulation amounts from the slot are temporarily shown on the controls.

Real time parameter modulation indicators

• All active modulation is animated in real time on the interface using a blue highlight shade, within either slider path or rotary outer rings. This can be seen on the left for Osc2's **Out** and **Filter Mix** parameters.

These indicators represent the modulation occurring on the last played voice.

Parameter to slot highlight

Hover the cursor over a parameter - any TransMod slots which are modulating the parameter are highlighted with a *bright orange* outline.

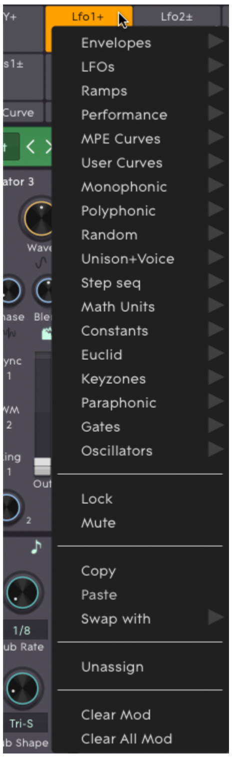
This screenshot shows the cursor hovered above Osc2's **Out** parameter to discover which slot(s) may be modulating it (slot 6 in this case).

Real time slot output indicators

As mentioned previously, the far right side of each TransMod slot features animated LED-style indicators representing its modulation signal output.

Managing TransMod slots

Cypher2 features a number of additional functions for managing TransMod modulation slots using the TransMod slot context menu.



These fuctions are available by right-clicking on the central part of a slot or on the upper/lower part (after the source/scale menu items).

Lock

This function locks the slot's contents from any changes other than direct modification - so preset-loading, Randomizer and morphing operations have no effect on the slot.

Mute

When activated, the **Mute** function stops the TransMod slot from having any effect. This can be very useful during sound design for analysing or modifying aspects of existing patches.

Copy/Paste, Swap With...

These functions allow the contents of TransMod Slots to be interchanged.

The **Copy** and **Paste** functions operate on the currently selected slot.

The **Swap With** function opens a sub-menu allowing the contents of the current slot to be exchanged with any of the other TransMod slots.

Unassign

Use the **Unassign** function to remove the source or scale function assignment from the TransMod slot. Any modulation depths for the slot are unaffected. If the slot is subsequently assigned to another TransMod source, the existing modulation depths become active again and target parameters respond to the newly assigned source.

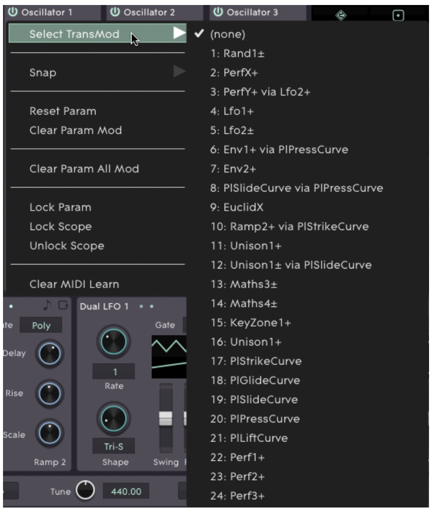
Clear Mod

The **Clear Mod** function clears any modulation depths specified for the TransMod Slot.

Clear All Mod

The **Clear All Mod** function clears all current modulations depths in all TransMod Slots.

Managing modulation with the Parameter context-menu



Right-click on a parameter to display its context menu, containing the following functions relating to the TransMod system:

Reset Param

The **Reset Param** function resets the control's initial value to its default setting.

Clear Param Mod

The **Clear Param Mod** function clears any modulation amount that exists for the control from the *current* TransMod slot.

Clear Param All Mod

The **Clear Param All Mod** function clears any modulation amounts that exist for the control from *all* TransMod slots.

Select TransMod

This sub-menu offers another way of selecting TransMod slots.

5.2 Modulator gating/triggering

The adjustable modulation sources shown on the Cypher2 interface - LFOs, Ramps, Mod Envelopes and Amp Envelope - are known as *gated modulators*.

This term refers to the fact that these modulators require an event - a 'gate' or 'trigger' - to function. A 'gate' can be considered as an on/off event message with a duration - essential for ADSR envelopes gated by MIDI note-on and note-off messages. On the other hand, Ramp modulators are 'triggered' with an 'on' message only - for example with MIDI note-on messages. While an LFO does not require a gate/trigger event to function, these events are used for retriggering or 'resetting' the LFO (returning it to the position dictated by its Phase control).

The most common gate/trigger source is MIDI note input. However, a number of additional gating/triggering modes are provided for more advanced patch design.

• *Dual LFO1 Gate modes*

Poly

The *Poly* setting is the default setting for all gated modulators in Cypher2. The modulator is gated or triggered with polyphonic MIDI note input.

When all available voices are exceeded, subsequent notes (which are played as a result of voice stealing) do not trigger the modulator unless the **Retrig** button is activated.

PolyOn

The *PolyOn* mode disregards the **Retrig** button setting entirely - the modulator is always gated or triggered by key-on events for each new voice that is generated.

Mono

In *Mono* mode, a gated modulator is gated/triggered by the first key-on when all notes are off – subsequent keys that are played without releasing all keys do not re-gate the modulator.

This behaviour occurs regardless of the current number of active voices.

Important note

It is important to be aware of the difference between the *Mono* gate mode and the **Mono** button on the LFO:

• Activating the **Mono** button forces the LFO to monophonic rather than polyphonic operation (in other words, a single LFO is available for all voices rather than an LFO being available per voice).

• In *Mono* gate mode, with the **Mono** button deactivated, the LFO is polyphonic but is only retriggered from its starting phase by the first key-on after all notes are off.

Song

The main function of the *Song* gate type is to provide a 'free-running' LFO which is not triggered by key-on events or other modulators.

When playback is started in the host/DAW, the LFO is triggered and runs freely afterwards. This allows free-running behaviour but with repeatability – each time the song is played the LFO runs identically.

When playback is stopped in the host, the LFO simply runs freely, continuously. In such a situation, it behaves just like a standalone synth that features a free-running LFO.

This gate mode can also be used for the Ramp, ModEnv and AmpEnv modulators. This is mainly to facilitate situations such as using the **Loop** envelope mode as a free-running LFO or to create a long Ramp that rises from 0 over the course of a song.

Please note that using this mode with the AmpEnv is a special case - while it is resynchronized by the Song phase, it is also always re-triggered by polyphonic note input in addition.

PolyKey

This mode is intended to be used when setting up duophonic/paraphonic sounds - it causes modulators to always retrigger when new MIDI notes are received.

When using paraphonic modulators, keyboard input is cycled 'within the voice'. Using the *PolyKey* gate mode can be especially useful with envelopes as they will also be retriggered within the voice, allowing the effect of the new note to be articulated.

See the Keytracking section later in this chapter for more details on using the *PolyKey* gate mode.

PolyOff

In *PolyOff* mode, the modulator is always triggered by key-off events for each voice. This is useful for release-stage effects such as generating a hammer-off sound via a ramp or envelope. It disregards the current **Retrig** button setting.

The *PolyOff* mode is very useful for the LIFT MPE dimension (note-off velocity) generated by the ROLI SeaBoard RISE and BLOCKS controllers, and can be used with any other keyboard which transmits note-off velocity.

A typical LIFT and *PolyOff* usage scenario:

• Set *PolyOff* as the Gate source for the Ramp section or one of the Mod Envelopes.

• Increase the Amp envelope **Release** parameter to create more time in which the release effect can occur.

• Modulate Ramp **Rise** or Mod Envelope **Release** via the *PlLiftCurve* TransMod source for a dynamic effect depending on the LIFT value.

• Create modulation from the relevant Ramp or Mod Envelope TransMod source - try the filter cutoff or FM amounts as a starting point.

• Try scaling this modulation via the LIFT or *OffVel+* (note-off velocity) TransMod sources, thus introducing another way of making the response more dynamic.

Other gated modulators

Gated modulators can be gated or triggered by a selection of other modulators – the gating or triggering occurs when the gate source reaches a value of 1.

For example:

• when setting an Envelope to be gated by the LFO, it is retriggered every time the LFO reaches its highest point (1).

• when setting the LFO to be gated by an Envelope, the LFO is retriggered (reset to the point specified by its **Phase** setting) every time the Envelope reaches its highest point (at the end of the **Attack** stage) .

5.3 Dual LFO

Cypher2's 2 Dual LFOs provide sources of movement for modulating parameters and can be particularly interesting when its own parameters are modulated. The **Shape** control offers many waveforms including several types of random shapes. The **PWM** and **Swing** controls provide additional control over the waveform shape.

The secondary 'sub LFO' is derived by clock-dividing the main LFO for additional movement rhythmically linked to the LFO. The sub LFO features its own **Sub Shape** control for setting its waveform.

Try using additive combinations of main and sub LFO TransMod sources to modulate the same parameters. The *LFO+Sub* source already combines these sources, but more flexible and complex effects can be achieved by using main and sub LFO sources in 2 TransMod slots. Modulate the **Sub Rate** and **Sub Shape** parameters to create complex and evolving new waveform shapes.

Each Dual LFO module can operate either polyphonically (per-voice) or monophonically (a single LFO for all voices as found in many vintage polysynths) depending on the state of its **Mono** button.

Gate modes

Each LFO features a number of Gate mode settings. For polyphonic 'key-on reset' behaviour, use the *Poly* gate mode. For 'free-running' operation, use the *Song* mode.

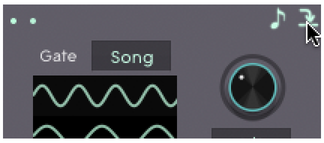
LFO1 features the *Poly*, *PolyOn*, *Mono*, *Ramp*, *Env1*, *Song* and *PolyKey* gate modes.

LFO2 features the *Poly*, *PolyOn*, *Mono*, *Ramp*, *Env2*, *Song*, *PolyKey* and *LFO1Sub* gate modes.

Main LFO controls

Rate / Sync ♪

The **Rate** parameter controls the rate or speed of the LFO's cycle.

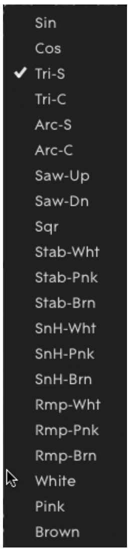


• *Setting up a free-running monophonic LFO with the* ***Mono*** *button and the* Song ***Gate mode***

If the **Sync** ♪ button is activated, BPM-based units are used when setting the **Rate**. With the **Sync** button deactivated, the **Rate** control is adjusted in Hz.

Mono

Enabling the **Mono** button forces the LFO into monophonic operation. In monophonic mode, there is only 1 global LFO for all voices instead of an LFO for each voice. With the button deactivated, the LFO is fully polyphonic - **Rate**, **Mode** and other parameters can all be distinctly modulated for each voice.



Mode

The **Mode** control selects the LFO shape and can be modulated via the TransMod system. Note that modulating this control continuously (as opposed to, say, with a key-on random or unison/voice source for example) is liable to result in abrupt jumps in the output which can cause audible clicks (depending on which destination parameters are modulated).

Standard LFO shapes

- Sine
- Cosine
- Tri-S
- Tri-C
- Arc-S
- Arc-C
- Saw-Up
- Saw-Down
- Square

Random Stab LFOs

- Stab-White
- Stab-Pink
- Stab-Brown

The various noise types provide random points to which the LFO 'jumps' on each cycle. After it reaches the next point, it drops down vertically to the position of the previous point. This process then repeats. The end result is a series of sharp 'stab' shapes.

Random Sample+Hold LFOs

- SnH-White
- SnH-Pink
- SnH-Brown

These modes are sample+hold values taken from noise sources. They create 'steppy' sounding random LFOs, because each value stays constant until the next value, at which point the LFO immediately travels vertically to the new value.

Random Ramp LFOs

- Ramp-White
- Ramp-Pink
- Ramp-Brown

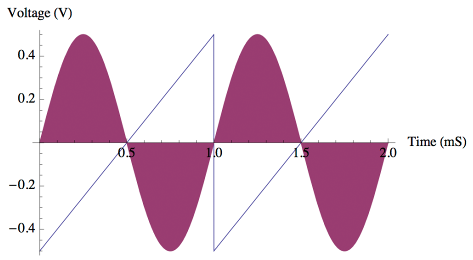
In these modes, the LFO ramps up or down to the next value – the end result is somewhat like a complex triangle shape but with random amplitudes.

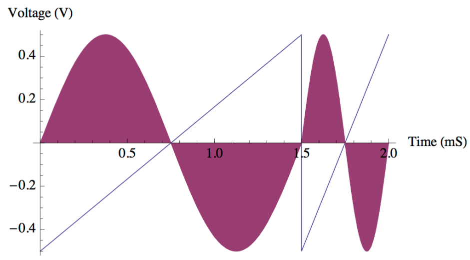
Noise LFOs

- White noise
- Pink noise
- Brown noise

These are straightforward noise sources for use as an LFO that output noise values continuously at control-rate.

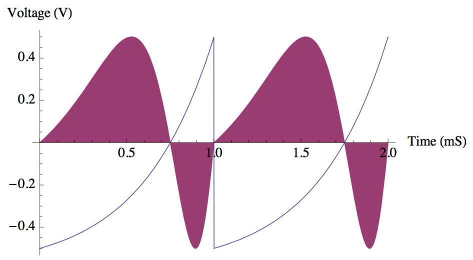
Swing





This is an unusual control for an LFO to possess. It creates a 'swung' feel by changing the length of each part of 2 LFO cycles so that each cycle has a different length, but their combined length remains the same.

PW (Pulse Width)



Again, this control is not often found on LFOs. This function changes the curve of each LFO cycle in a similar way to changing the pulse width on a pulse oscillator. It works on all shape modes.

The diagram illustrates the function at a setting of 75% on a sine shape.

Phase

The **Phase** parameter allows the phase of the LFO to be adjusted between 0 and 360 degrees. Phase can be modulated in real time, and always resets when the LFO is gated.

Sub LFO controls

Sub Rate

The Sub LFO offers a divided or multiplied version of the LFO - it is always synchronized to the LFO but allows additional modulation variation which is rhythmically linked to the LFO.

It can also have an entirely different waveform shape (specified by the **Sub Shape** parameter).

The Sub LFO can run at the following ratios of the LFO speed:

1/8	*1/3*	*1*	*2*	*4*
1/6	*1/2*	*1 1/3*	*2 1/3*	*5*
1/5	*2/3*	*1 1/2*	*2 1/2*	*6*
1/4	*3/4*	*1 2/3*	*3*	*8*

Sub Shape

The **Sub Shape** control specifies the waveform shape of the Sub LFO - the available shapes are identical to those for the main LFO's **Mode** control.

LFO quadrature outputs

The provided set of TransMod LFO sources include bipolar quadrature outputs for each LFO at phase offsets of 90°, 120°, 180°, 240° and 270° relative to the main bipolar output of the LFO (LFO +/-) at 0° phase.

An additional bipolar Sub LFO output is provided with a 90° phase offset relative to the main bipolar Sub LFO output.

Quadrature LFO sources can be used for 'barber-pole' style effects with multiple oscillators but can be used for time-locked movement for all kinds of parameters. For example, try modulating FM amounts, waveform shapes, osc levels, wavefolder gains, filter cutoffs and effects parameters.

TransMod sources (LFOs menu)

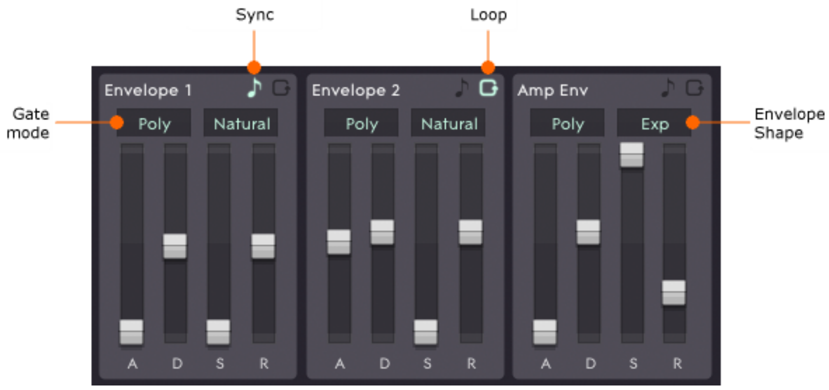
LFO1$^{+}_{-}$	LFO 1 main out (bipolar)
LFO1+	LFO 1 main out (unipolar)
LFO1_Sub	LFO 1 main out (bipolar)
LFO1Sub+	LFO 1 main out (unipolar)
LFO2$^{+}_{-}$	LFO 2 main out (bipolar)
LFO2+	LFO 2 main out (unipolar)
LFO2_Sub	LFO 2 main out (bipolar)
LFO2Sub+	LFO 2 main out (unipolar)
LFO1_90	LFO 1 main out (90° phase)*
LFO1_120	LFO 1 main out (120° phase)*
LFO1_180	LFO 1 main out (180° phase)*
LFO1_240	LFO 1 main out (240° phase)*
LFO1_270	LFO 1 main out (270° phase)*
LFO1_Sub_90	LFO 1 sub out (90° phase)*
LFO2_90	LFO 2 main out (90° phase)*
LFO2_120	LFO 2 main out (120° phase)*
LFO2_180	LFO 2 main out (180° phase)*
LFO2_240	LFO 2 main out (240° phase)*
LFO2_270	LFO 2 main out (270° phase)*
LFO2_Sub_90	LFO 2 sub out (90° phase)*

* all bipolar

TransMod sources (Gates menu)

LFO1Gate+	Gate output (effectively a clock output) of LFO 1 square wave
LFO2Gate+	Gate output (effectively a clock output) of LFO 2 square wave

5.4 Amp and Mod Envelopes



An envelope allows a sound to be shaped over time, from its beginning to its end. It is sometimes called a 'contour'.

Amp Envelope

The **Amp Envelope** is directly routed internally to the Amp parameter in Cypher2 with a fixed depth of modulation. This is to avoid having to make a dedicated modulation connection between them purely for engaging the amp so that notes can be heard.

The TransMod system allows the Amp Envelope to be routed to almost all synth parameters.

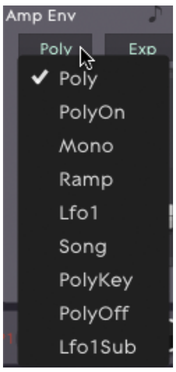
Envelope 1 & 2

Envelope 1 and **Envelope 2** can be used for a variety of purposes, although they do feature depth-controlled direct routings to the **VCF 1+2** parameter (which controls the cutoff of each filter simultaneously when its **Link** button is activated). They can be considered similarly to the second envelope typically found on many monophonic and polyphonic synths, which is often routed to the filter cutoff.

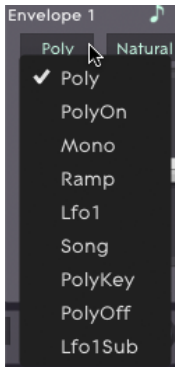
The TransMod system allows these envelopes to be routed to almost all synth parameters.

Gate modes

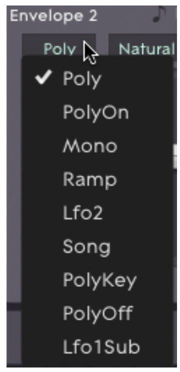
Each envelope features the following Gate mode settings:



The Amp Envelope features the *Poly*, *PolyOn*, *Mono*, *Ramp*, *Lfo1*, *Song*, *PolyKey*, *PolyOff* and *Lfo1Sub* gate modes.



Envelope 1 features the *Poly*, *PolyOn*, *Mono*, *Ramp*, *Lfo1*, *Song*, *PolyKey*, *PolyOff* and *Lfo1Sub* gate modes.

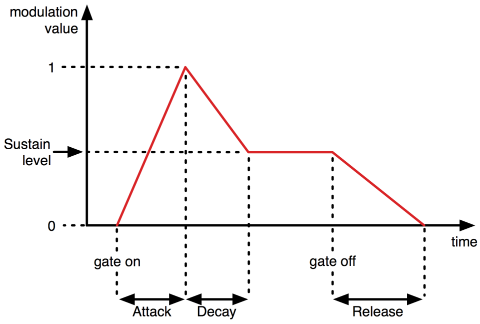


Envelope 2 features the *Poly*, *PolyOn*, *Mono*, *Ramp*, *Lfo2*, *Song*, *PolyKey*, *PolyOff* and *Lfo1Sub* gate modes.

ADSR controls

A (Attack) / D (Decay) and **R (Release)** are periods of time, while **S (Sustain)** is a level, expressed in %.

When the envelope is gated, the following processes occur:



• The envelope level rises from 0 to 1 over the defined **Attack** time

• After this has been reached, it decays towards the level defined by the **Sustain** control, over a time period defined by the **Decay** parameter (if **Sustain** is at 100%, there is effectively no Decay stage).

• This occurs as long as the envelope is gated (while the gate signal is 'on'). Whenever the gate is released (when the gate signal returns to an 'off' state), no matter which stage has been reached, the envelope's level falls to 0 over the time defined by the **Release** parameter.

All of these parameters can be modulated with the TransMod system.

Sync ♪

With the **Sync** ♪ button activated, the time-based envelope controls (**Attack**, **Decay**, **Release**) are set in BPM units.

With the Sync button deactivated, these controls operate in seconds.

Loop

Activating the **Loop** button causes the envelope to repeat after it has completed the **Attack** and **Decay** phases. After the Decay time is complete, the envelope starts again - this continues while the envelope is gated. If the **Sustain** control is higher than 0, the envelope travels towards the sustain level and remains at this level for the duration of the Decay time.

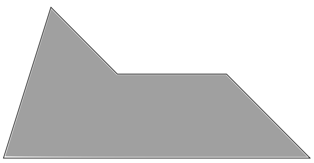
When the gate is released, the level of the looping Attack and Decay stages is scaled down to 0 over the **Release** time.

The level of the envelope returns to 0 when it is gated, and at the end of each 'loop' (after each decay stage).

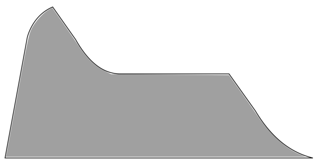
The Loop function is useful for simulating echo/delay effects on gate release or for providing an additional LFO capable of alternative shapes.

Envelope shape

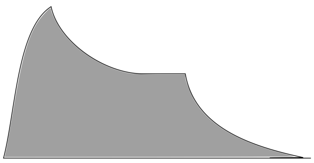
The **Envelope Shape** menu provides a selection of responses which each produce a different shape. Each is useful for different types of sounds.



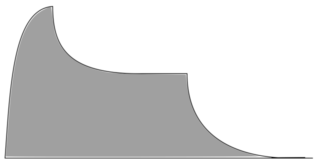
Linear



Natural



Cubic



Exp (Exponential)

Certain parameters such as **Amp**, **Pitch**, **Scale** and other tuned parameters like filter **Cutoff** controls are often more suited to *Linear* or *Natural* envelopes in many situations. The *Natural* type very closely resembles the *Linear* shape but with a more natural, less 'abrupt', end to the **Decay** and **Release** stages.

Please note that although the **Envelope Shape** menu sets the 'main out' of the Envelope, all shapes are generated and available simultaneously as TransMod sources.

Also try modulating an Envelope's **ADSR** stage parameters by its own final output - this can be useful in creating even more extreme envelope curves, or for dynamically changing the final output shape.

TransMod sources (Envelopes menu)

Env1+	Envelope 1 main out
Env1-	Envelope 1 main out (inverse)
Env1Lin+	Envelope 1 Linear out
Env1Nat+	Envelope 1 Natural out
Env1Cub+	Envelope 1 Cubic out
Env1Exp+	Envelope 1 Exponential out
Env2+	Envelope 2 main out
Env2-	Envelope 2 main out (inverse)
Env2Lin+	Envelope 2 Linear out
Env2Nat+	Envelope 2 Natural out
Env2Cub+	Envelope 2 Cubic out
Env2Exp+	Envelope 2 Exponential out
AmpEnv+	Amp Envelope (positive)
AmpEnv-	Amp Envelope main out (inverse)
AmpEnvLin+	Amp Envelope Linear out
AmpEnvNat+	Amp Envelope Natural out
AmpEnvCub+	Amp Envelope Cubic out
AmpEnvExp+	Amp Envelope Exponential out

5.5 Ramp Generators

Ramp generators are versatile modulators which are mainly intended for use with the TransMod system. They can be particularly useful as TransMod scale functions - for example, to delay and/or fade in an LFO. They can also be used as a trigger source for envelopes or as a delayed retrigger source for LFOs, using the Gate source selector for these modulators.

When the Ramps section is gated (in fact, when it is triggered, as it ignores duration/note-off), the following occurs for both of the 2 Ramp generators:

• Its value drops immediately from its maximum value of 1 to 0 (minimum value).

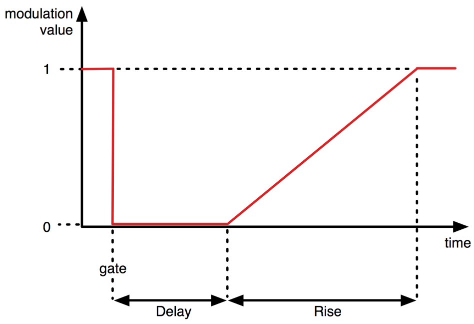
• Then, after a period of time defined by its **Delay** parameter (during which its value remains at 0), it rises to 1 over a period defined by its **Rise** parameter.

Gate modes

The Ramp features the *Poly*, *PolyOn*, *Mono*, *Lfo1*, *Song*, *PolyKey*, *PolyOff* and *Lfo1Sub* Gate mode settings.

Delay

The **Delay** parameter sets the delay after the ramp value drops from 1 to 0, and before it begins rising to 1. It is scaled by the value of the **Scale** control.



Rise

The **Rise** control adjusts the time taken for the ramp to rise from 0 to 1. It is scaled by the **Scale** control.

Scale

This control is a multiplier for the Ramp's **Delay** and **Rise** times. By default, it is set at 100%, meaning that the Delay and Rise times occur exactly as set on the respective controls. The **Scale** control multiplies these times between 0% and 200%.

Sync ♪

With the **Sync** ♪ button activated, both Ramp generators' **Delay** and **Rise** times are set in BPM units. With the Sync button deactivated, these parameters are set in seconds.

Loop

With the **Loop** button activated, both Ramp generators repeat their **Rise** period indefinitely after the **Delay** period is completed once. It therefore provides a useful additional polyphonic LFO-style modulator with a saw-wave shape.

Tips for using Ramp generators

• The inverted Ramp TransMod sources - *Ramp-* and *RampTrans* - provide simple linear and cubic hold-decay envelopes which ignore duration/note-off and have zero attack. The **Delay** parameter provides the hold phase and the **Rise** parameter provides the decay or release phase.

• A *Ramp+* source as a scale function can provide a delay and fade-in for an LFO source - set the **Delay** and **Rise** parameters as desired.

• With the **Loop** button activated, the Ramp generators provide polyphonic triggered LFO-style modulators with a saw-wave shape and optional initial delay. Try using a Curve processor to transform it into a different shape.

• Try using *Ramp1* as the gate source for an envelope - the envelope is now delayed by Ramp1's **Delay** time.

• Ramps are useful in conjunction with the LIFT MPE curve (or note-off velocity) - use *PolyOff* as the Ramps' gate source, then set up TransMod modulation from a *Ramp-* or *RampTrans* source to parameters as desired, with the LIFT curve as a scale function (the *PlLiftCurve* source). This provides a 'stabby' envelope shape for the release phase, the amount of which is dictated with note-off velocity.

TransMod sources (Ramps sub-menu)

Ramp+	Ramp 1 (positive)
Ramp-	Inverted Ramp 1 This is very useful as a triggered envelope shape (it is not gated like an envelope).
RampTrans	This is a version of *Ramp-* but with a slewing function applied which smooths its shape to a curve.
Ramp2+	As above for Ramp 2.
Ramp2-	
Ramp2Trans	

5.6 Keytracking

Keytracking is a polyphonic modulation source which simply represents pitch input from the keyboard. Unlike the more dynamic forms of modulation such as envelopes and LFOs, it does not feature any adjustable parameters - keytracking involves each MIDI key sending a static control signal, with each sequentially ascending key sending out a higher control signal than the last. It can be considered in a very similar way to calibrated '1 volt per octave' keytracking CV signals in modular systems.

Each key pitch received by Cypher2's MIDI input is passed as a pitch control signal to the next available voice. These signals persist for the voice until there are no more available voices and existing voices are 'recycled' via voice-stealing - the control signals for newly-received notes override those for existing voices according to the voice **Priority** setting.

Keytracking performs several basic functions for which Cypher2 provides dedicated controls:

Oscillator pitch tracking

• In order to play an oscillator 'musically' with the keyboard, its **Key** button must be activated. This sets a 100% keytracking amount, meaning that each ascending key increases the pitch by a semitone.

Filter tracking

• In order to play the filter musically (especially useful with the **Resonance** control turned up so that the filter self-oscillates), set its **Keytrack** depth control to 100%.

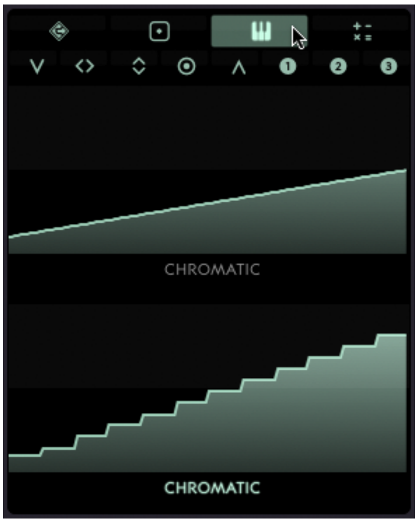
• A filter **Keytrack** setting of 100% is also important for creating the same timbral response with the filter relative to keyboard pitch.

TransMod source (Polyphonic menu)

As well as the direct routings and depth controls specified above, keytracking can also be achieved for any other parameter by using the *Pitch* source in the TransMod system.

Pitch$^{+}_{-}$	Tracking source derived from keys played on the keyboard, measured in octaves: increases by 1 for each octave going up the keyboard. Modulate pitch-based parameters by 12 semitones for full tracking.

Keyzone1 & Keyzone2



Although the *Pitch*$^{+}_{-}$ TransMod source can be used for keyboard tracking on any parameter, there may be situations which demand a more flexible keytracking response. For these situations the 2 Keyzone processors can be used. Both these ; processors provide a graphical editor for defining an adjustable keytracking response - an arbitrary curve can be drawn.

KeyZone processors are similar in concept to Curve processors but are designed especially for keyboard tracking purposes. They also feature discrete steps rather than the vector spline curve found in Curve processors.

The KeyZone editors are shown in the Scope panel - click the **Keyzones** button to display them.

The default curve for KeyZone1 is identical to that of the *Pitch* TransMod source - a linear response. Click/drag on the curve display in order to draw a customized response.

- Hold down SHIFT while click/dragging to draw straight lines
- Hold down CTRL and SHIFT to draw a straight horizontal line

KeyZone2 is set to Chromatic mode by default, with the **Chromatic** button activated. This is a special mode which allows each note in the 12-tone scale to be assigned a discrete modulation value. The KeyZone2 sources can then be used in the TransMod system in order to create modulation which results in each note of the scale sounding different.

KeyZone TransMod sources (Keyzones menu)

KeyZone1+	The output of the Keyzone1 editor
KeyZone1$^{+}_{-}$	As above but bi-polar
KeyZone2+	The output of the Keyzone2 editor
KeyZone2$^{+}_{-}$	As above but bi-polar

ScaleNote TransMod source (Keyzones menu)

The *ScaleNote+* source provides a way of tracking notes in an octave of the 12-tone scale. It ranges from 0 at C to 1 at B, increasing by $^{1}/_{12}$ for each note.

ScaleNote+	C=0, increasing by $^{1}/_{12}$ for each semitone up to B=1, regardless of which octave is played

Paraphonic and Duophonic keytracking sources

Cypher2 provides paraphonic keytracking sources for recreating classic 2-osc (duophonic) or 3-osc (paraphonic) effects. Such techniques are typically found on some monophonic synths as a means of creating pseudo-polyphonic sounds using multiple oscillators through a single filter.

While Cypher2 is a fully polyphonic synth, paraphonic techniques can create unique effects which are interesting to play, especially when using ring-mod, FM and other modulation between the oscillators.

The 'D' (Distance, or interval) paraphonic sources effectively allow each osc's pitch to be tuned on the fly with keyboard input, sequentially - before the next actual voice is triggered and the process begins again. Additional sources are provided for distributing the same velocity values to all voices ('monophonic' velocity operation) and also simple logic outputs (0 or 1) from the lowest/highest/oldest/newest note.

Paraphonic TransMod sources (Paraphonic menu)

DLowest+ DHighest+ DOldest+ DNewest+	These sources provide the intervals to the lowest, highest, oldest and newest active incoming notes. Intervals are measured in octaves - therefore, pitch-based destination parameters should be modulated by 12 semitones for conventional musical use.
VelOldest+ VelNewest+	These sources allow monophonic velocity modulation for destinations. They send the oldest or newest velocity value received to all voices.
IsLowest+ IsHighest+ IsOldest+ IsNewest+	These sources return a value of 1 for the lowest/highest/oldest/newest note, allowing parameters to be modulated discretely for these notes. For example, modulating an osc's Out control with *IsLowest+* is useful for introducing it only for the lowest note, leaving higher notes unaffected - meaning it can be used as a bass note osc.

3-osc paraphonic operation example

Try the following steps to start to explore paraphonic modulation:

1. In the default preset, select *DLowest+* and *DHighest+* as the sources in 2 TransMod slots.

2. Right-click on the Osc2 and Osc3 **Scale** controls and click *Equal* in the Snap sub-menu.

3. Increase the **Out** sliders for Osc2 and Osc3 so they can be heard.

4. Select the slot with the *DLowest+* source and set up a modulation depth of +12 on Osc2's **Scale** parameter.

5. Select the slot with the *DHighest+* source and set up a modulation depth of +12 on Osc3's **Scale** parameter.

Now play and hold down multiple notes - you will hear paraphonic effects:

• When only 1 key is held down, the 3 oscs are set to the same pitch.

• When 2 keys are held, the lowest of the notes retunes the Osc2 pitch while Osc1 and Osc3 remains the same.

• With 3 notes held, the lowest note retunes Osc2 and the highest note retunes Osc3, while Osc1 remains the same.

Try varying combinations of the various paraphonic sources for different playing responses!

Using the *PolyKey* gate mode for paraphonic envelope operation

When using paraphonic modulation, the *PolyKey* **Gate mode** can be useful for envelope retriggering for each key.

Following on from the previous example:

6. Deactivate Filter 1's **Link** button, select a TransMod slot with the *Env1+* source and create modulation on the Filter 1 **Cutoff** parameter.

7. Play and hold down multiple keys - you will notice the envelope's effect is heard only on the first note.

8. Now set Envelope 1's **Gate mode** to *PolyKey* and try again - the envelope is retriggered on each note.

9. Now try it with the Amp Envelope - decrease the **Sustain** parameter and set its **Gate mode** to *PolyKey*. Now the amplitude of each new paraphonic osc combination is articulated with the Amp Envelope when each new incoming note occurs.

5.7 Velocity

Velocity is perhaps the most immediate form of performance control that can achieve expressive results polyphonically (for each voice). It involves transmitting a value depending on the speed or intensity with which each key is played, between 1 (softest playing intensity) and 127 (highest playing intensity).

Note-on and note-off velocity

Almost all MIDI keyboards are capable of transmitting *note-on velocity* values. While *note-on velocity* is the most commonly used form of velocity control, Cypher2 also supports *note-off velocity*, whereby a keyboard transmits a value based on the speed/intensity of the key *release.*

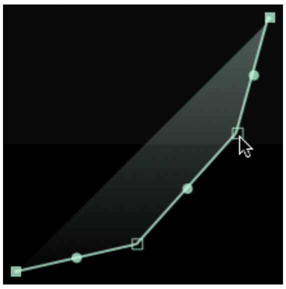
Many keyboards are incapable of transmitting note-off velocity. Such keyboards typically send out a note-on velocity value of 0 upon a key being released in order to stop a note that is playing. While this is a perfectly valid method of ending the duration of a note in the MIDI protocol, it is not capable of producing the dynamic effects on release which can be achieved with note-off velocity.

Velocity TransMod sources (Polyphonic menu)

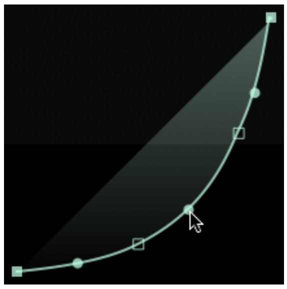
OnVel+	Note-on velocity
OffVel+	Note-off velocity

Using note-on velocity with the MPE STRIKE ∨ curve

The STRIKE ∨ Curve processor in Cypher2's Scope panel allows note-on velocity response to be adjusted with an arbitrary vector spline curve, allowing MPE velocity response to be tailored to each preset.



• Click and drag each of the 4 square nodes to set the basic shape of the output.



• Click and drag the circular handles up/down to adjust the curve between the 4 square nodes.

Right-click on the Curve editor to display a **Reset** function allowing the curve to be reset to the default linear shape.

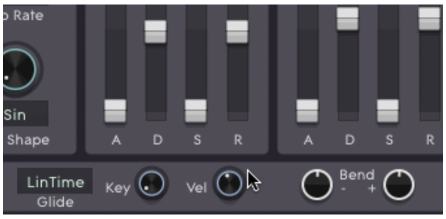
Note-off velocity and the MPE LIFT ∧ curve

The LIFT ∧ MPE curve provides a similar function to the STRIKE ∨ curve, except for note-off velocity rather than note-on velocity.

MPE Velocity TransMod sources (MPE curves menu)

PlStrikeCurve	STRIKE (velocity processed through STRIKE curve)
PlLiftCurve	LIFT (note off velocity processed through LIFT curve)

Velocity Glide



The **Velocity Glide** control creates glide times to new notes' velocity values, allowing a further expressive element to performances.

When a note velocity value is received when a key-on event occurs, this special glide function smooths out over time any transitions to the modulation depths defined in slots driven by the *OnVel+* or STRIKE sources.

It can be considered in a similar way to regular pitch glide which smooths transitions to new pitch values, but it acts on any synthesis parameter to which velocity modulation has been assigned.

Patching ideas for STRIKE/note-on velocity and LIFT/note-off velocity

• Modulate the **Amp** parameter with *OnVel+* or STRIKE in order to vary amplitude with velocity.

• Try using *OnVel+* or STRIKE as TransMod scale functions in existing slots - for example, try scaling envelope modulation to dynamically vary the effect of the envelope depending on playing velocity.

• *OnVel+* / STRIKE can be very effective for producing dynamic behaviour when modulating envelope parameters such as **Attack**, **Decay** and **Release**.

5.8 Monophonic sources

Monophonic TransMod sources operate globally on all active voices - unlike polyphonic sources which modulate each voice independently.

Typical examples are continuous MIDI messages such as MIDI CCs, mono pressure (channel aftertouch) or mod wheel - these all represent a single stream of signals which are distributed to all active voices.

With the exception of the monophonic sources described below, Performance controller sources and monophonic gate/trigger sources, all other TransMod sources described in this chapter are polyphonic.

Common MIDI controller sources (Monophonic menu)

These sources are provided to facilitate direct control of a TransMod slot from common MIDI CCs. Also see the next section on performance macro controls .

Mod+	Modulation wheel: MIDI CC #1
Breath+	Breath controller: MIDI CC #2
Expr+	Expression: MIDI CC #11
MnBend$^{+}_{-}$	Pitch bend (bipolar, octaves)
MnBend+	Pitch bend (unipolar, octaves)
MnPress+	Channel pressure (monophonic aftertouch)

Song phase sources (Monophonic menu)

These sources provide a saw-up ramp synchronized to the host tempo. Usually, song phase is used for mono free-running gated modulators with repeatable behaviour, but they are also provided in the TransMod system for extra mono LFOs synced to 1 or 4 beats of the host tempo.

Phase1$^{+}_{-}$	Song phase: a saw-up ramp from 0 to 1 per beat, based on song position
Phase1+	As above, but uni-polar
Phase2$^{+}_{-}$	Song phase: a saw-up ramp from 0 to 1 over 4 beats, based on song position
Phase2+	As above, but uni-polar

Step-sequencer sources (Step seq menu)

Cypher2's Sequencer page features modulation lanes that each correspond to a monophonic TransMod source (available in uni-polar and bi-polar versions).

As the sequence is stepped through by the timing clock the specified values for each step are output to these TransMod sources.

StepSeq$^{+}_{-}$	Step-sequencer main (bipolar).
StepSeq+	Step-sequencer main (unipolar).
StepSeq1+	Step-sequencer 1 (unipolar)
StepSeq1$^{+}_{-}$	Step-sequencer 1 (bipolar)
StepSeq2+	Step-sequencer 2 (unipolar)
StepSeq2$^{+}_{-}$	Step-sequencer 2 (bipolar)
StepSeq3+	Step-sequencer 3 (unipolar)
StepSeq3$^{+}_{-}$	Step-sequencer 3 (bipolar)

Numeric constants (Constants menu)

Numeric constants are very useful scale functions for the modulation from a TransMod slot.

0, 0.1, 0.5, 1, 2, 10, -1	1 is the default scale function for all TransMod slots, meaning that each source is unchanged before modulating the destination parameters.

5.9 Macro controls and Euclid processor

Macro / Performance controllers (Perf1 / Perf2 / Perf3 / Base X / Base Y)

Macro controls are generic interface elements which are used to modulate actual synthesis parameters via the TransMod system.

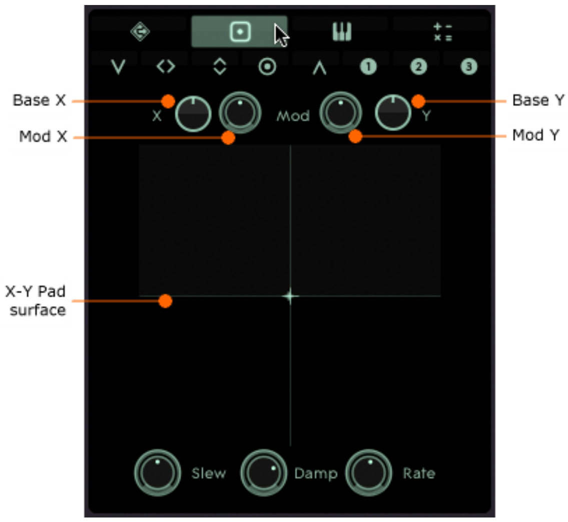
They act monophonically – they send the same control values to all voices - and are intended to be mapped to monophonic MIDI continuous controllers (CCs) using the built-in MIDI Learn functionality. Typically MIDI CCs are transmitted by controls such as modulation wheels and additional generic knobs/ sliders on MIDI controller devices.

The macro controls include the **P1**, **P2** and **P3** controls at the lower-right of the interface as well as an X-Y pad (**Base X** and **Base Y**) in the Euclid processor (see below).

Performance TransMod sources (Performance menu)

Perf1+	Performance controller 1
Perf2+	Performance controller 2
Perf3+	Performance controller 3
PerfX+	This source outputs the **Base X** value (without any further Euclid processing)
PerfY+	This source outputs the **Base Y** value (without any further Euclid processing)

Euclid processor



The Euclid processor utilizes Euclidean geometry and a spring model to provide an interesting way of creating chaotic and complex modulation signals for use in the TransMod system. It can also be used as an X-Y pad (mapped to an external X-Y pad controller if desired) with adjustable inertia.

The Euclid processor editor is shown in place of the Visualizer Scope/Curve/Keyzone processors - click the **Euclid** button to display it.

The X-Y position is set by the **Base** and **Mod** controls while the **Slew**, **Rate** and **Damp** functions dictate the way that movements to new positions occur.

Base X
Base Y

These controls set the base position on the X-Y area - they can be adjusted either by moving the rotary controls themselves or by clicking anywhere on the X-Y area. **Base X** and **Base Y** are very similar to the **Perf1** / **Perf2** / **Perf3** controls - they are intended to be monophonic performance controllers and are perfect for assigning to a hardware X-Y pad controller. In fact, their direct signals are available with the PerfX+ and PerfY+ TransMod sources - these output the Base X and Base Y values without any of the Euclid processor's additional functions being applied.

Mod X
Mod Y

The **Mod X** and **Mod Y** controls apply an offset to the X and Y axes and can be modulated using the TransMod system in order to create new modulation signals by processing existing TransMod sources - the resulting new modulation shapes are available with the Euclid sources shown below.

Slew

This control adjusts the amount of **Slew**, or lag, applied to movement of the X-Y position.

Rate

The **Rate** control adjusts the speed of movement of the X-Y position.

Damp

The **Damp** control adjusts the amount of damping applied to the X-Y position's inertia. The inertia is derived from a spring model - increasing the Damp control 'tightens' the spring so that less elasticity is evident for X-Y position movements.

Euclid TransMod sources (Euclid menu)

EuclidX	X-axis output from Euclid processor
EuclidY	Y-axis output from Euclid processor
EuclidRadius	The distance from the centre of the X-Y area to the current value
EuclidAngle	The angle from the centre of the X-Y area to the current value

X/Y TransMod sources (Performance menu)

PerfX+	This source outputs the **Base X** value (without any further Euclid processing)
PerfY+	This source outputs the **Base Y** value (without any further Euclid processing)

Examples and ideas for the Euclid processor

• Creating chaotic LFOs

Modulate the **Mod X** control with the LFO+ source and modulate the **Mod Y** control with the LFOSubUni source (with the Sub LFO's division set to 1/3 for example). Use the *EuclidRadius* source to modulate parameters with the result.

• Introducing slew or elasticity to any TransMod modulation source

Slewing TransMod sources can be achieved with the Curve processors, but can also be performed in the Euclid processor if required. Using the **Damp** control can produce new modulation shapes with an elastic 'bounce'.

Modulate the **Mod X** or **Mod Y** parameters with any TransMod source, increase the **Slew** and **Damp** controls (while also modulating these parameters over time if desired) and use the *EuclidX* or *EuclidY* TransMod sources to modulate parameters with the result.

For example, using a PlGate+ or PlOnGate+ source can result in a slow ASR envelope shape suitable for a pad sound. The envelope response can be further refined with the **Rate** and **Damp** controls.

Try processing the following sources in the same way:

• the LFO set to one of its random shapes - this can produce interesting new LFO shapes

• one of the *RampTrans* or *Ramp-* sources for a rounded envelope

• a Ramp source with the Ramp section's **Loop** button activated for interesting poly LFOs

5.10 Voice and Unison sources

Voice and Unison sources allow polyphonic modulation effects which spread parameter values within the modulation depth range according to the number of active voices. Active voice counts are converted into a set of values from -1 to 1 (bi-polar) and 0 to 1(uni-polar). Each source is provided in 2 different varieties of value distribution between the minimum and maximum.

The Unison sources allow any parameter to be varied across unison voices as opposed to most vintage synths which were limited to detuning the pitch of unison voices although some examples (such as the Oberheim OB-8) featured trim pots to adjust the pan position of each voice.

Unison and Voice sources in the TransMod system enable effects more reminiscent of the original Oberheim polysynths based on multiple discrete SEM modules, which allowed the programming of a totally different patch for each voice.

Note that polyphonic Random sources are also very useful for per-voice modulation of parameters - subtle values can simulate slight imperfections between voice cards in an analogue polysynth; extreme values can be great for experimental new sounds.

Voice count TransMod sources (Unison+Voice menu)

Voice1$^{+}_{-}$	Each active voice generates a value proportionally distributed between -1 and 1
Voice1+	As above but uni-polar
Voice2$^{+}_{-}$	As Voice1 but distributed in a different order
Voice2+	As above but uni-polar

Unison TransMod sources (Unison+Voice menu)

The Unison sources allow any parameter to be varied across unison voices – not just the pitch detuning found in classic synths.

Unison1$^{+}_{-}$	Each active unison voice outputs a value proportionally distributed between -1 and 1 This source is especially suited to panning voices across the stereo field.
Unison1+	As above, but uni-polar
Unison2$^{+}_{-}$	As Unison1 but distributed in a different order This distribution can be more suitable for unison detuning with osc pitch modulation.
Unison2+	As above, but uni-polar

Alternating voice TransMod sources (Unison+Voice menu)

The Alt sources output alternating values for each voice used in sequence:

Alt1$^{+}_{-}$	Alternating source per key-on: -1, 1, -1, 1...
Alt1+	As above but positive only: 0, 1, 0, 1...
Alt2$^{+}_{-}$	Alternating source per key-on: -1, 0, 1, -1, 0, 1...
Alt2+	As above but positive only: 0, 0.5, 1, 0, 0.5, 1...

Examples and ideas for using voice and unison TransMod sources

• Modulate the **Pan** control for a stereo spread of voices during chords or unison notes.

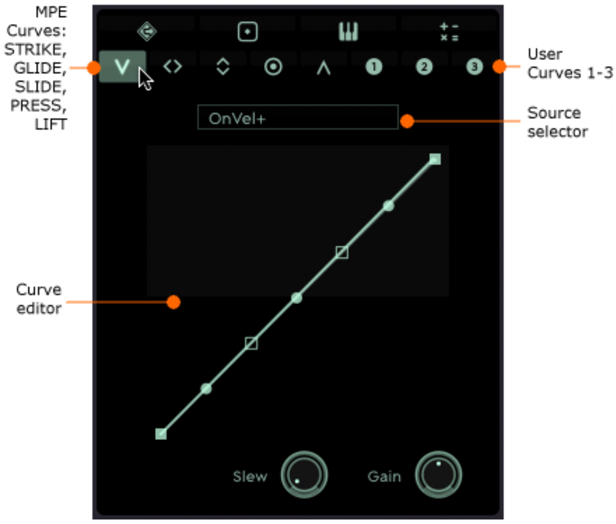
• Subtle voice and unison modulation on envelope, filter, osc **Pitch** and other controls can result in more variation during performances and emulate the slightly uneven, unpredictable feel of a vintage synthesizer containing voice cards with slight differences in calibration or pitch-tracking.

• Wide pitch modulation (for example, a large amount of modulation on the Coarse master pitch control) with Unison sources can be very effective for synthesis of cymbals, percussion sounds other types of atonal/noisy/abstract atmospheric timbres.

• Modulate **Pitch Glide** and **Velocity Glide** controls to vary glide times for each voice.

• Use the Voice/Unison sources as TransMod scale functions to inject per-voice variation for other modulation sources.

5.11 Curve processors



Cypher2 features a number of curve processors which allow incoming modulation to be remapped to a different output curve. All Curve processors operate polyphonically so can be used for per-voice modulation. Specifying a monophonic TransMod source results in monophonic output from the Curve processor.

Curve processor editors are shown in the Scope panel, accessed by the buttons shown on the left.

MPE Curves: ∨ <> ◇ ⦿ ∧

5 of the available 8 curve processors are dedicated to MPE controller usage allowing, for example, velocity and pressure signals to be remapped from the default linear response to a different response using a 3-segment spline vector shape.

The MPE Curves process 'raw' MPE MIDI messages from their default linear response to an arbitrary user-defined response curve:

PlStrikeCurve	Output from STRIKE ∨ Curve Processor acting on the OnVel+ source.
PlGlideCurve	Output from GLIDE <> Curve Processor acting on the PlBend+ source. For ROLI hardware, the recommended modulation amount from this source is +48 on the Tuning section's **Pitch** parameter. It is also possible to create the same depth on individual oscillator sections' **Scale** controls instead.
PlSlideCurve	Output from SLIDE ◇ Curve Processor acting on the PlSlide+ source.
PlPressCurve	Output from PRESS ⦿ Curve Processor acting on the Press+ source.
PlLiftCurve	Output from LIFT ∧ Curve Procrssor acting on the OffVel+ source.

Please note that, if desired, these may be remapped to process any other TransMod sources - click the Curve Source selector to display a menu of all available TransMod sources.

User Curves ❶ ❷ ❸

In addition, 3 freely assignable curve processors are also provided, each allowing any TransMod modulation source's response to be remapped.

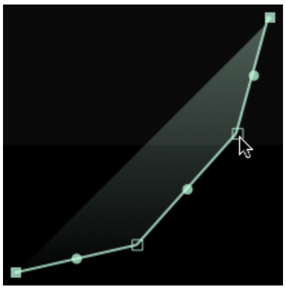
PlU1Curve	Output from User Curve Processor 1 ❶: assigned to act on the Perf1+ source by default
PlU2Curve	Output from User Curve Processor 2 ❷: assigned to act on the Perf2+ source by default
PlU3Curve	Output from User Curve Processor 3 ❸: assigned to act on the Perf3+ source by default

Curve Source selector

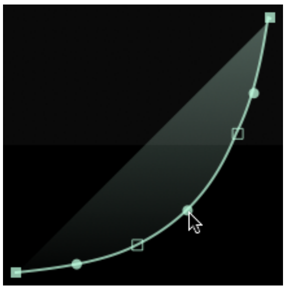
Click the Source selector to display a menu containing TransMod modulation sources. Navigate to and click the required source to specify it as the input source for the Curve processor.

Note that the sources for all curves, including dediated MPE curves, can be changed if desired. Please take care when changing MPE curve input sources if you intend your sounds to be used by other creators.

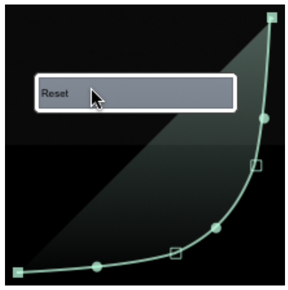
Curve Editor / Reset



Click and drag each of the 4 square nodes to set the basic shape of the output.



Click and drag the circular handles up/down to adjust the curve between the 4 square nodes.



To reset the curve to the default linear shape, right-click in the Curve Editor and use the **Reset** function.

Reset

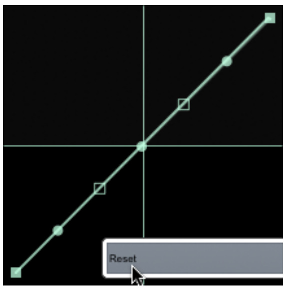
Right-click on the curve graph to display the Reset function. This reverts the curve's shape to its default linear setup, discarding any edits. The curve's 3 adjustable nodes are spaced equally apart, except if the Bipolar button is activated on the Glide curve: in this case, the first curve node is placed exactly in the centre position, allowing easy setup of a 'mirrored' bipolar curve shape.

Bipolar (MPE GLIDE <> and User Curves ❶ ❷ ❸ only)

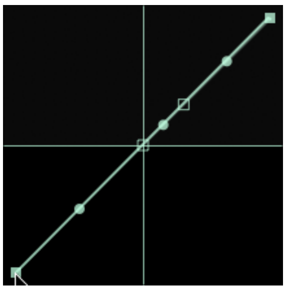
By default all curves' output values range from 0 (minimum) to 1 (maximum).

The GLIDE <> and User curves ❶ ❷ ❸ allow output values ranging from -1 to 1 if the **Bipolar** button is activated.

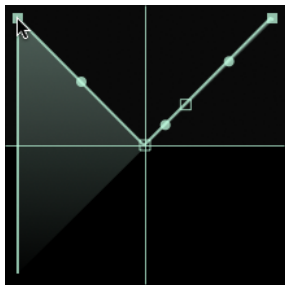
For the GLIDE curve, this mode is provided in order to facilitate a 'mirrored' GLIDE response - in other words, the same response if the GLIDE direction is up or down.



1. Use the **Reset** function while **Bipolar** mode button is active.



2. Note that the 2nd square node is placed in the centre.



3. Now the 1st square node can be moved to the top to create a mirrored response in both directions.

Slew

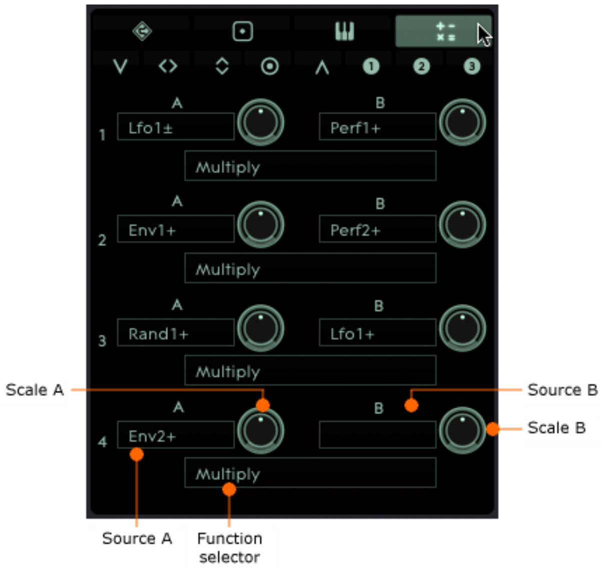
This parameter introduces a variable amount of lag in the output curve. A small amount of **Slew** is recommended for MPE controller dimensions in order to smooth out any abrupt jumps in the incoming values. Larger amounts lead to a noticeable lag in the output curve response.

Gain

The **Gain** control variably scales the curve output from 0 (no curve output) through 1 (the default setting - no scaling applied) to 2 (the output is doubled).

5.12 Maths processors

Cypher2's Maths processors provide the capability for complex modulation processing by applying mathematical functions to multiple TransMod modulation sources.



They can be used for creation of fascinating, complex modulation shapes and are also very useful for balancing the effect of cumulative modulation of parameters from multiple sources. The latter can be very useful when creating '5D' MPE sounds.

The Maths editor is shown in the Scope panel - click the **Maths** button to display it.

Each of the 4 available Maths processors provides the following parameters:

Source A

Click the currently selected source to display a menu containing TransMod modulation sources. Navigate to and click the required source to specify it as **Source A** for the Maths processor.

Scale A

This control provides a multiply function which variably scales **Source A** before the maths function is applied, from 0 through 1 (the default setting - no scaling applied) to 2 (the source is doubled).

This control can be modulated by a TransMod modulation slot, effectively meaning that **Source A** can be multiplied by 2 multiplied TransMod sources within the slot.

Source B & Scale B

These controls specify the same as above for the second input source for the Maths processor.

Function selector

This menu specifies the mathematical function to be applied to Sources A and B. By default, the *Multiply* function is selected. However a range of other mathematical functions are available.

TransMod sources (Math units submenu)

Maths1$^{+}_{-}$	Output from Math Unit 1 (bipolar)
Maths2$^{+}_{-}$	Output from Math Unit 2 (bipolar)
Maths3$^{+}_{-}$	Output from Math Unit 3 (bipolar)
Maths4$^{+}_{-}$	Output from Math Unit 4 (bipolar)

5.13 Oscillator sources

Oscillator sources (for LFO mode use)

Each of the 3 oscillators are provided as TransMod sources in order to modulate parameters when using an osc's Low Frequency button. Of course it is also possible to use the FM 2 / FM 3 / WM 2 controls on the interface itself for modulating frequency and waveshape on their respective oscs when using oscs 2 or 3 in LFO mode.

Please note that, even though these sources will be active regardless of the state of the Low Frequency button on an osc, they are quantized to control-rate like all other TransMod modulation sources.

TransMod sources (Oscillator menu)

Osc1$^{+}_{-}$	Oscillator 1 (control-rate)
Osc2$^{+}_{-}$	Oscillator 2 (control-rate)
Osc3$^{+}_{-}$	Oscillator 3 (control-rate)

Beat frequency rates

The beating rates between the 3 oscillators within a voice are provided as TransMod sources.

These sources are active regardless of the state of the constant-beat detuning effect of using an osc's **Fine** control:

• Using the **Fine** controls causes beat effects which remain constant across the keyboard range.

• Beating is also caused when detuning using the **Scale** controls - in such cases the beating rates vary across the keyboard.

The Beating TransMod sources allow other parameters to be modulated in parallel with the beating effects.

TransMod sources (Oscillator menu)

Beat12$^{+}_{-}$	Beating (Oscs 1+2, bipolar)
Beat21$^{+}_{-}$	Beating (Oscs 2+1, bipolar)
Beat23$^{+}_{-}$	Beating (Oscs 2+3, bipolar)
Beat32$^{+}_{-}$	Beating (Oscs 3+2, bipolar)
Beat31$^{+}_{-}$	Beating (Oscs 3+1, bipolar)
Beat13$^{+}_{-}$	Beating (Oscs 1+3, bipolar)

5.14 Random sources

The TransMod system contains a range of polyphonic random modulation sources of various types, from simple noise sources to more specific ways of generating random values. With the exception of the Noise sources (the first 4 entries below), each of the random sources acts on key-on events: these are not constantly changing/evolving random sources. For the latter, the LFO with its various random modes should be used or, alternatively, creative use of the Euclid and Curve processors.

Random TransMod sources (Random menu)

Noise1$^+_-$	Fast-varying white noise random value
Noise1+	As above, but uni-polar
Noise2$^+_-$	Fast-varying pink noise random value
Noise2+	As above, but uni-polar
Rand1$^+_-$	A single random value for all destination parameters, generated by key-on sample-and-hold on white noise
Rand1+	As above but uni-polar
Rand2$^+_-$	A single random value for all destination parameters, generated by key-on sample-and-hold on pink noise
Rand2+	As above but uni-polar
MRand1$^+_-$	Individual random values for all destination parameters (multiple random values), generated by key-on sample-and-hold on white noise. This source is useful for modulating each note with a different random value every time it is played.
MRand1+	As above but uni-polar
MRand2$^+_-$	Identical to MRand1$^+_-$ but using pink noise
MRand2+	As above but uni-polar
VMRand1$^+_-$	A random number per voice, per parameter, that is generated at synth load time, generated using white noise. This source is useful for having a set of random values that stays constant throughout the current session.
VMRand1+	As above but uni-polar
VMRand2$^+_-$	Identical to VMRand1$^+_-$ but using pink noise
VMRand2+	As above but uni-polar
Drift$^+_-$	Slowly changing random LFO This can be used to simulate subtle parameter drift in hardware analogue synths ñ for example. small amounts of modulation on osc pitch can simulate drifting VCOs.

5.15 Gate/Trigger sources

Monophonic gate/trigger sources (Gates menu)

Normally, gates and triggers are used for various types of gated modulator behaviour which is specified outside the TransMod system (using the drop-down menus on each gated modulator).

However, they are also provided as TransMod sources for complex modulation operations (try using them as scale functions or within Maths processing), or as a way of achieving quick pulses for filter stabs and so on without having to use an envelope.

Gate and trigger sources simply output a value of 1 or 0 depending on several conditions.

MnGate+	Outputs 1 if one or more keys are held down, otherwise 0
MnOnGate+	Outputs 1 if one or more keys are held down, but is retriggered on every key-on (output returns to 0 and immediately to 1)
MnTrig+	Outputs 1 then immediately 0 on key-on when a note is played after all notes are off
SongGate+	Outputs 0 if the host song is not playing, and 1 if it is playing: this gates on the nearest beat

Polyphonic gate/trigger sources (Gates menu)

These are very similar to the monophonic gate/trigger sources, except that they operate polyphonically (they can occur multiple times depending on the number of available voices).

PlGate+	Outputs 1 if one or more keys are held down, otherwise 0 Retriggering occurs according to the state of the Retrig button (returns to 0 and immediately to 1).
PlOnGate+	1 if one or more keys are held down, but retriggers on every key-on (returns to 0 and immediately to 1)
PlTrig+	This is the same as PlGate+, except that it outputs a 1 and then returns to 0 immediately when keys are played.

Clock sources (Gates menu)

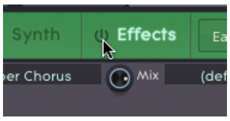
These clock-type sources are continuous streams of simple pulses. They are provided as bonus sources for use in the TransMod system especially for use with modulation processing in the Maths and Euclid modules.

Clock96ppq+	This is the clock signal used for tempo-synced functions in Cypher2 - it is also provided within the TransMod system as a bonus source for experimental purposes, especially for modulation processing in the Maths processors. In the plugin this clock signal is derived from the host's tempo; when using the standalone version of Cypher2, it is derived from the **Tempo** parameter in the Sequencer page.
Lfo1Gate+	Gate output (effectively a clock output) of LFO 1 square wave
Lfo2Gate+	Gate output (effectively a clock output) of LFO 2 square wave

6 Sequencer page

The Sequencer page provides advanced step sequencer and arpeggiator functions which allow interesting ways of performing with sounds. Click the **Sequencer Page** button in Cypher2's Navigation bar in order to access it.

Sequencer Page Power ⏻



The **Sequencer Page Power** ⏻ button provides a global activation function for the entire Sequencer section.

With the button deactivated, the entire Sequencer section is inactive.

Master section

This set of controls dictates how the Sequencer page is used.

Rate

The **Rate** control sets the speed of the clock used for the Step Sequencer - it is set as a rhythmic division of the **Tempo**.

Tempo

When running Cypher2 as a plugin within a DAW/host, it is not possible to adjust the **Tempo** control during playback as Cypher2 is locked to the host's tempo when the transport is running. However, when the host transport is stopped, this control can be adjusted to any desired tempo although it will be resynchronized to the host tempo when the transport is next started.

Please note that this control cannot be modulated by the TransMod system.

Mode

This control dictates whether the Sequencer page is set to modulation-only functionality (*Off*), *Arpeggiator* mode or *Step Seq* mode.

With this control set to *Off*, the Step Sequencer will not generate any notes to feed into Cypher2's synth engine. Instead, it simply allows the Sequencer lanes to be used as TransMod modulation sources.

Trig

With the **Trig** button activated, the Step Sequencer is reset to step 1 when the keyboard is used for either Arpeggiator or Step Seq Mode.

Input

When the **Input** button is activated, keyboard input is routed to the Arpeggiator and Sequencer. With the button deactivated, all keyboard input is routed directly to the Cypher2 synthesis engine, although the Sequencer may still be operational in addition.

Record

While the **Record** button is activated, the Sequencer does not play through its steps - instead, it waits for incoming MIDI notes, with each being recorded as sequential steps within the main Step Sequencer and Gate Sequencer.

When the **Record** button is deactivated, the Step Sequencer and Gate Sequencer **Length** is automatically set to the number of notes received and the recorded sequence is ready to play using any of the **Gate** / **Hold** / **Latch** settings

Gate / Hold / Latch

Only one of these buttons can be active at any one time:

Gate

In this mode, the Sequencer must be played by holding down MIDI notes with the **Input** button activated.

Hold

This mode is similar to **Gate** mode except the Sequencer holds any MIDI notes received to keep the Sequencer playing indefinitely. Again, the **Input** button must be activated for the Sequencer or Arpeggiator to receive note input.

Latch

Latch mode does not require keyboard input in order to play the sequencer: instead, it is constantly played, and plays from the first step when the host transport when it is started (when using Cypher2 as a plugin). To stop playback of the Sequencer, switch back to **Gate** mode or deactivate the **Sequencer Page Power** button.

Step Sequencer section

The main Step Sequencer stores root note and relative note interval values from -24 to +24 semitones. Values can either be painted in directly on the interface or played in using the **Record** function (which also generates corresponding Gate Sequencer events).

The root note is set as the first note received during a **Record** operation. However, when using the mouse/trackpad to simply paint values directly onto the interface, the root note is set to C.

Please note that the actual notes heard as a result are also dependent upon the other functions in the Sequencer page, such as transposing with keyboard input, the Matrix section's **Pitch** parameter and the Scale Processor section.

Right-click in the Step Sequencer area to **Copy** / **Paste** the sequence between the 8 available **Memory** slots (described in the Matrix section below).

Length

The **Length** parameter sets the number of active steps for the Step Sequencer. This parameter is also set when using the **Record** button - it is set to the number of notes received during the Record operation.

Gate Sequencer section

The Gate Sequencer provides events that articulate voices in the Cypher2 synthesis engine. It runs in parallel to the Step Sequencer and is locked to the same **Length** setting, although using the **FIL** Gate Seq Mode and the Matrix section, it is possible for one of the Mod Sequencers to create a gate sequence with a different number of steps.

Right-click in the Gate Sequencer area to show a **Reset** function, reverting to the default setup.

Gate seq modes

Each step in the Gate Sequencer can be set to a different type of gate behaviour. Click a step to show a menu with the following Gate Seq modes:

[.] (Off)

The step is not played - it instead acts as a 'rest' step.

[■] (On)

The step is always played.

[FIL] (Fill)

The step is played if the Matrix section's Variable input value is higher than 50%. This can be set using the Variable parameter (either manually or via TransMod modulation) or by selecting one of the 3 Mod Sequencers by changing the *Off* setting to *1*, *2* or *3*.

When using the latter, the Mod Sequencer's **Length** setting is effectively used as the length of the Gate Sequencer, with 'ON' steps being dictated by Mod Sequencer step values above 50.

[PRB] (Probability)

The step is played with a probability determined by the Matrix section's **Variable** input value.

[ALT]

The step is played once in every *X* cycles, with *X* determined by the Matrix section's **Variable** input value.

[FOL] (Follow)

The step follows the play status of the previous step in the sequence.

[NOT] (Not Follow)

The step's play status is the inverse of that of the previous step in the sequence.

Copying a Gate seq mode across multiple adjacent steps

While holding down the SHIFT key, click and drag right from any step to paint its Gate seq mode to adjacent steps as required.

Arpeggiator section

To use the Sequencer page in Arpeggiator mode, adjust the Master section's **Mode** control to the *Arpeggiator* setting - without this setting the following controls will have no effect. This setting repurposes the Sequencer engine to act as a classic arpeggiator function which converts held MIDI notes to generate arpeggio sequences on the fly. Please note that the **Input** button in the Master section must be activated in order to route MIDI input to the Arpeggiator function.

Mode

This setting, in conjunction with the **Priority** setting, dictates how keyboard notes are arranged for the Arpeggiator sequence.

Fwd

Notes are arranged according and played according to the Priority setting. After the sequence completes, it starts again from the beginning.

FwdRev

In the *FwdRev* Mode, the Arpeggiator sequence is arranged according to the Priority setting, plays forward as in the Fwd mode and then plays in reverse before starting again from the beginning.

Rnd

In *Rnd* Mode, notes are arranged in a random order to form the Arpeggiator sequence.

Manual

The *Manual* Mode uses each Step Sequencer value to dictate which note in the Arpeggiator sequence should be played. The value range in each Sequencer step is divided by the number of notes in the Arpeggiator sequence - each step's value plays the corresponding note in the Arpeggiator sequence.

Range

By default, the Arpeggiator sequence is created from the notes played from the keyboard. The **Range** control allows the Arpeggiator sequence to be extended by repeating it over ascending octaves, up to a maximum of 4 octaves.

Priority

The **Priority** control dictates the order in which keyboard notes held down simultaneously are inserted into the Arpeggiator sequence - the sequence plays *towards* the specified setting.

Newest

The *Newest* note (that is, the last played note), is always placed at the end of the Arpeggiator sequence. This setting creates the Arpeggiator sequence according to the order in which notes are played.

Oldest

The *Oldest* note is always placed at the end of the Arpeggiator sequence.

Highest

The *Highest* note is always placed at the end of the Arpeggiator sequence - this setting always results in classic 'ascending' arpeggios.

Lowest

The *Lowest* note is always placed at the end of the Arpeggiator sequence - this always results in descending arpeggios.

Hardest
Softest

These settings use the velocity of input MIDI notes to dictate the order of the Arpeggiator sequence. In *Hardest* mode, the highest velocity note is placed at the end of the Arpeggiator sequence, while in *Softest* mode, the lowest velocity note is always placed at the end of the Arpeggiator sequence.

Mod Sequencer 1, 2 & 3

The 3 Mod Sequencers run in parallel with the main Step Sequencer. They can be used either as individual TransMod modulation sources to animate parameters over time, or can be used in conjunction with the Matrix controls in order to influence the main Step Sequencer and Scale Processor modules.

The 3 available Mod Sequencers can be used in both the following ways:

• as additional modulation sequencer sources for use with TransMod modulation

• in conjunction with the main Step Sequencer and Gate Sequencer for more advanced functionality using the Matrix section and Gate Seq modes

Rate

Each Mod Sequencer's **Rate** control sets the speed of its clock. It is set as a rhythmic division of the **Tempo**. but it can be an entirely different setting than the main Step Sequencer's Rate.

Length

Each Mod Sequencer also has its own **Length** setting which dictates the number of steps in its sequence - again, it can be entirely different to that of the main Step Sequencer.

Memory

Each of the Mod Sequencers possess 8 Memory slots, only 1 of which can be currently active depending on the setting of the **Memory** control.

The Memory contents comprise only the step values for all 32 steps - other settings like **Length** and **Rate** are not included.

Slew

The **Slew** control applies an adjustable amount of lag, or glide, between values output by the Mod Sequencer to its relevant TransMod source.

Matrix section

The Matrix section provides a number of advanced parameters for introducing variation into sequences and allowing Mod Sequencers to interact with the Step Sequencer.

Each of the following parameters can:

- be adjusted directly with its rotary control
- have its rotary control modulated by the TransMod system
- be played via keyboard input (**Pitch** and **Velocity** parameters only)
- additionally be set to be modulated by one of the 3 Mod Sequencers using the **Source Select** buttons

Shift (Step & Fine)

The **Step Shift** parameter offsets the start point of the sequence by up to 32 steps in either direction.

The **Fine Shift** parameter can be considered as a 'micro-shift' - it offsets the currently active step by up to 1 step in either direction, effectively allowing syncopation and other sub-step timing effects.

Pitch

The **Pitch** control offsets, or transposes, the pitch of notes transmitted by the Step Sequencer to the Cypher2 synth engine. Transposition of up to 24 notes in either direction can be specified.

Note that transposition is also dictated by keyboard input.

Transposition occurs before **Scale** processing is applied.

Duration

The **Duration** parameter sets the length of gate events transmitted by each step to the Cypher2 synthesis engine.

Velocity

The **Velocity** parameter sets the velocity of notes transmitted to the Cypher2 synth engine. This will dictate how any velocity-based TransMod modulation is affected.

Key Velocity

The **Key Velocity** button underneath the **Velocity** control routes velocity values from keys played to the notes transmitted from sequencer events to the Cypher2 synth engine.

Variable

This parameter is used in conjunction with the FIL and PRB Gate Sequencer modes.

Memory

The Step Sequencer and Gate Sequencer feature 8 Memory slots, only 1 of which can be currently active depending on the setting of the Matrix section's **Memory** parameter.

Scale

The **Scale** parameter selects the active Scale slot from the 8 slots available in the Scale Processor section.

Scale Processor section

The Scale processor section provides 8 separate Key and Scale 'pitch-quantizer' slots from which to select with the Scale control in the Matrix. This Scale parameter can be modulated over time using TransMod modulation or influenced by any of the 3 Mod Sequencers.

Key

This control displays a menu allowing any Key from C to B to be selected for each of the 8 available Scales.

Scale

This control allows a corresponding scale for each of the 8 Scale slots to be selected.

The default is Chromatic, meaning that no scale processing takes place and sequencer notes are unaffected.

Transpose from [note]

Activating this function sets an interval from which to transpose to the new scale. It is typically more useful when using sequences in a single key - in such situations it can effectively help to easily transpose into different keys.

It may be less relevant when working with sequences in multiple keys in multiple Memory slots.

For example, for a sequence originally in C major, and setting a slot's **Key**/**Scale** as *F Major*:

• with the **Transpose from** button deactivated, events are transposed to the nearest notes in the F major scale

• with the **Transpose from** button activated, events are first transposed up a 4th (from a key of C to F) before the scale quantization is applied, meaning that the melody is preserved while being transposed to the new key.

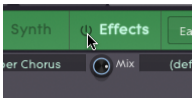
7 Effects page

Using Cypher2's Effects section

Cypher2's Effects section is located on the Effects page - click the **Effects** button on the interface in order to access it.

Effects Page Power ⏻

The **Effects Page Power** ⏻ button provides a global activation function for the entire Effects section - with the button activated, the Effects section is also activated.



With the button deactivated, the entire Effects section is bypassed.

FX A and FX B Chains

Cypher2's Effects section comprises the FX A and FX B Chains which are arranged in series with FX A feeding into FX B, unless the **Parallel** button is activated (see below). Each chain features 3 FX Slots for inserting available FX devices.

FX A Mix & FX B Mix

Each FX Chain features its own mix control that can be modulated by the TransMod system. It blends between the input and the FX Chain's output.

The **FX A Mix** control blends between Cypher2's raw output and the output of the FX A chain.

The **FX B Mix** control adjusts the blend between the output of the FX A and FX B chains.

Parallel mode

With the **Parallel** button activated, the FX A and FX B effects chains operate in parallel rather than in series (A into B).

When activating the Parallel function, the **A-B Route** parameter must be used in order to route any signal into the B effects chain.

The A-B Route parameter can be modulated polyphonically to vary the proportion of each voice between the A and B chains. If one of the chains is left empty, the control can effectively act as a polyphonic effect send per-voice to the other chain. In such scenarios try modulating the A-B Route control with velocity or with an MPE source, perhaps via a ramp, envelope or LFO to make things more interesting.

FX Slot controls

Each of the 3 FX Slots in each FX Chain features the following common controls:

Power ⏻

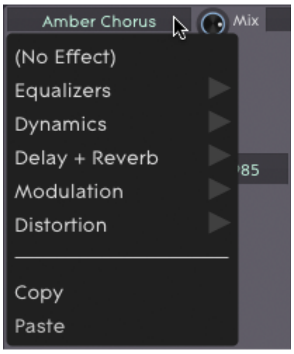
The Slot's **Power** ⏻ button activates/deactivates the device loaded into the FX Slot.

Mix

Each FX Slot features a **Mix** control for blending between the incoming signal (*0*) and the fully-processed output (*100%*) from the device loaded into the FX Slot.

Device menu

This drop-down menu selects an FX device to load into the FX Slot.



• *Device menu*

It also contains functions to **Copy** and **Paste** the contents of the Slot.

Preset menu

The **Preset menu** provides a drop-down menu of the available presets for the device.

Prev / Next preset ◂ ▸

The **Prev** ◂ and **Next** ▸ buttons cycle through the available presets. The currently loaded preset name is shown on the **Preset menu**.

• *Preset menu*

Reordering FX Chains

The sequence of FX devices within the 6 Slots can be changed using drag and drop. To move an FX device, simply drag it to the desired Slot - other devices are rearranged accordingly within the available Slots.

It is also possible to use the **Copy** and **Paste** functions on each Slot's **Device menu** in order to copy FX devices with their settings to other Slots.

Effects and TransMod modulation

Cypher2's FX device parameters can be modulated with the TransMod system in a similar way to the Synthesis engine controls.

Effects and polyphony

The crucial difference when modulating the Effects section is that it is *monophonic* - when using more than 1 voice in the Cypher2 Synth section, the multiple voices are mixed down to a stereo signal and then fed into the Effects section which then acts on the entire signal.

When using *polyphonic* TransMod modulation sources, the Effects section applies the modulation signals from the *last voice played*. In other words, it features *Newest* note priority.

7.1 EQ

EQ

The EQ device provides 4 EQ bands including two parametric mid bands.

L (Low) and H (High) bands

The Low and High bands are switchable between a bell with a fixed **Q** of 2.5 octave (with the **Bell** button activated) and a shelving response with a fixed rolloff of 12 dB per octave (with the **Bell** button deactivated).

• Low band **Freq** range: 40 to 600 Hz

• Low band **Gain** range: +/- 16.5 dB

• High band **Freq** range: 600 Hz to 14 kHz

• High band **Gain** range: +/- 20 dB

LM (Low Mid) and HM (High Mid) parametric bands

The Mid bands feature bell curves with adjustable Q.

• Low Mid Band **Freq** range: 200hz to 2 kHz

• High Mid band **Freq** range: 800hz to 7 kHz

• **Q** range: 0.5 octave to 2.5 octave

Band Power

Each band has its own **Power** button - turning off a band reduces CPU usage.

EQ-6F

The EQ-6F device is similar to the EQ device with a few differences:

• 4 parametric mid bands instead of 2

• High-pass and Low-pass filters as well as Shelf/Bell Low and High bands.

Pitch EQ

The Pitch EQ is tuned to musical notes and features 3 EQ bands including a parametric mid band.

Mix Filters

The EQ-Filter effect provides non-resonant high-pass and low-pass filters for broad tonal shaping of signals.

HP Frequency

The -18dB/oct high-pass filter removes frequencies below the specified frequency which can be set within a range between 20 Hz and 500 Hz.

LP Frequency

The -12 dB/oct low-pass filter removes frequencies above the specified frequency, which can be set within a range of 35 Hz to 3 kHz.

Power buttons

The **HP In** and **LP In** buttons are power buttons for each filter – when activated, the filter is enabled, and when deactivated, the filter is bypassed.

Enhancer

The Enhancer device is intended to add high-frequency harmonics to emphasize these frequencies in the signal. It is useful for adding 'sizzle' to sounds to help them cut through dense mixes.

Frequency

The Enhancer adds harmonics at and above the **Frequency** defined with this control.

Mode

The **Mode** control adjusts how the Enhancer circuit reacts to transients in the incoming signal. *I* is the slowest setting while *IV* is the fastest.

Amount

The **Amount** control adjusts the level of the harmonics which are added to the signal before the output.

7.2 Dynamics

Comp Chan

Comp Chan (channel compressor) is derived from a classic 'feedback-based FET limiting amplifier' design.

It features a 'fixed-threshold' design – the threshold at which compression starts is not adjustable. In practice, this means that the **Input** and **Output** controls may need adjustment when changing ratios.

Input / Output

The **Input** control adjusts the level of the signal entering the compressor. Once the input level has reached the internal threshold, compression begins. When this happens, use the **Output** control to turn down the increased input. The **Input** control ranges from -20 to +40 dB, while the **Output** control ranges from -40 dB to +20 dB.

Attack

The **Attack** control has a range between 0.02 ms and 1.2 ms.

Release

The **Release** control has a range between 50ms and 1.2 seconds.

Ratio

This control sets the compression **Ratio** to 4:1 , 8:1 , 12:1, 20:1 or 'Nuke', which is an emulation of the 'all buttons' mode on a classic limiting amplifier design. It results in a particularly brutal type of compression with accompanying distortion artifacts.

Comp Bus

This device is based on a classic bus compressor design from the centre section of a well-known British large-format mixing console.

It is most commonly used to add 'glue' and power to a drum or mix bus. However, it also works very well as an instrument compressor in its own right and offers a different flavour of compression to that of the Comp Chan.

Key HP (Key signal High-pass)

The **Key HP** control adjusts a variable high pass filter on the input signal that is used for the compressor's amplitude detection.

No filtering is applied to the audible signal - only to that being used to drive the peak detection circuit.

This control is useful when there is too much low-end in the signal resulting in the compressor reacting more heavily than desired.

Attack

Six **Attack** times are available: 0.1 ms, 0.3 ms, 1 ms, 3 ms, 10 ms, 30 ms.

Release

Five **Release** settings are available: 0.1 ms, 0.3 ms, 0.6 ms, 1.2 ms and Auto.

Ratio

Three **Ratio** settings are available: 2:1, 4:1 and 10:1.

Threshold

Unlike the Comp Chan, the Comp Bus device allows the **Threshold** to be adjusted. The Threshold represents the signal level at which the compressor begins to react.

Output

The **Output** control allows the overall output level to be increased after the compressor circuit has applied gain reduction to the input signal.

Limit

The **Limit** button applies analogue non-linearities to the input to the compressor's amplitude detection circuit (while not affecting the input signal itself). This results in a more transparent character to the compression effect, especially on attack phases of transients.

CompVCA

The CompVCA device is a modelled emulation of a VCA-based compressor circuit with a fast and clean compression characteristic.

Input

The **Input** control adjusts the signal level entering the compressor circuit. Higher levels result in the compressor circuit reacting more heavily.

Attack

The **Attack** time can be set between 100µs (microseconds) and 100ms (milliseconds).

Release

The **Release** time can be set between 10ms and 2s.

Ratio

The **Ratio** can be set between 1:1 and 50:1.

Threshold

The **Threshold** represents the signal level at which the compressor begins to react.

Output

The **Output** control allows you to increase the overall output level after the compressor circuit has applied gain reduction to the input signal.

NoiseGate

A noise gate is a type of dynamics processor that attenuates the input signal until its amplitude exceeds an adjustable threshold level, at which time the gate 'opens' to allow audio through at its actual level.

It can be useful for 'chopping' effects with a sound whose amplitude is being modulated by an LFO, or for helping to create 'gated reverb' effects.

When setting very fast **Attack** and **Release** times, it is common to hear 'clicks' in the audio when the gate opens and closes, especially with sounds predominantly comprised of low frequencies such as kicks and toms. This behaviour is completely normal - these times simply need to be increased slightly to overcome the problem.

Attack

The **Attack** control adjusts the speed at which the gate opens once the **Threshold** has been exceeded by the input signal's amplitude.

Hold

The **Hold** parameter controls the amount of time the gate remains open after the input signal has dropped below the **Threshold** level.

Release

The **Release** control adjusts the speed at which the gate closes at the end of the hold time.

Threshold

The **Threshold** control sets the level at which the gate starts to open. When the input signal amplitude exceeds the level specified by the Threshold control, the gate starts to open to allow audio through.

LPF and HPF Input Filters

This control provides high-pass (**HPF**) and low-pass (**LPF**) filters to process the input signal used to trigger the gate while leaving the actual processed signal *unfiltered*. This allows certain frequencies in the input to be isolated to improve the gating response – for example, excessive low frequencies in the input can make the gate react more sensitively than required.

Hysteresis

Noise gates have a tendency to open and close very quickly when the input signal's amplitude remains close to the threshold level for longer periods, something that can result in 'gate chatter'.

Increasing the **Hysteresis** control smooths out the gate response to reduce this problem, although the gate becomes less sensitive to small changes around the threshold level.

EnvShaper

Env Shaper offers an alternative approach to dynamics processing by adjusting the intensity of attack and sustain portions of transients.

Attack

The **Attack** control adjusts the intensity of the attack phase of detected peaks in the audio signal. Increase the control to intensify attack transients, and decrease it to soften transients.

Sustain

The **Sustain** control adjusts the intensity of release portions of detected peaks in the audio signal, which increases or decreases the apparent sustain of sounds in the signal.

Increase this control for more sustain, and decrease it for less sustain.

Adjusting the Sustain can be useful for creative emphasis or de-emphasis of reverb effects earlier in the FX chain.

Signal Bias

The **Signal Bias** control adjusts the sensitivity and release characteristics of DCAM EnvShaper. At low settings it is more sensitive to short transients while at higher settings it is more sensitive to longer transients.

Gain

The Gain device is a simple tool for increasing or decreasing a channel's level, useful as a final make-up gain or attenuation stage before Cypher2's output.

Gain Amp

The **Gain** control increases the channel's gain up to 18 dB, or decrease it up to -inf dB.

Linear Amp

This control provides a linear-response attenuator for the channel's gain - at a setting of 100% the signal is unaffected while at a setting of 0% the signal is fully attenuated.

Pan

The **Pan** control adjusts the balance between the left and right channels in the stereo field.

7.3 Delay and Reverb

Delay

Time

The delay **Time** can be set either in absolute time values in seconds or in tempo-based values (depending on the the **Sync** setting).

Sync

With the **Sync** button activated, the delay time is set in tempo-based values relative to Cypher2's current tempo. Possible values range from 64th note to 2 bars, including dotted and triplet variations. With the button deactivated, the delay time is set as an absolute time value, ranging from 31 ms to 4 seconds.

Swing

This control opens up a wide variety of delay grooves. The Delay effect features 2 taps, which are played at the same time with the swing control at the centre position.

By turning down the **Swing** control, the left tap is moved before the right tap up to a distance of half the delay time. By turning up the control, the right tap is moved up to a distance of half the delay time before the left tap.

Feedback

The **Feedback** control sets the amount of delay regenerations, caused by feeding the delayed signal back into the input. Higher values feed the delayed signal back in at a higher amplitude, leading to more regenerations of the input signal. Settings of 100% lead to indefinite regenerations until the value is reduced, and with good use of the built-in filtering can result in classic psychedelic, dubby analogue delay effects.

Hi (High-pass) / Lo (Low-pass)

These controls provide high-pass (**Hi**) and low-pass (**Lo**) filtering to each delay regeneration.

Dirty Delay

This device is very similar to the regular Delay effect but it is in fact an emulation of a variable-samplerate delay engine, using very similar algorithms to those found in the DirtyDAC effect. This type of engine was typically used in early digital delays and shares many sonic characteristics with analogue BBD delays.

The **Base Time** parameter sets the effective length of the delay's RAM buffer and the samplerate changes according to both the **Time** parameter (the actual desired delay time) and the Base Time (the length of the RAM buffer).

The **Bits** and **Anti-alias** parameters operate identically to the corresponding controls in the DirtyDAC device.

FX-Verb

FX-verb is a high-end algorithmic reverb device.

Pre-delay

The **Predelay** control introduces a delay between the dry sound and the reverberated output, creating a sense of space and distance.

Room Size

This control adjusts the size of the virtual reverberation chamber. Increasing the **Room Size** leads to a more pronounced and longer reverberation effect.

Decay

This control adjusts the length of the reverberation tail (which is also affected by the **Room Size** control).

Density

The **Density** control adjusts the density of reflections in the generated reverberation.

Damping Gain

The **Damping Gain** control adjusts the amount of damping applied by the Damping EQ before the output of the FXverb device. Increasing this control leads to more attenuation at the **Damping Freq**.

Damping Freq

The **Damping Freq** control adjusts the centre frequency of the Damping EQ positioned before the final output.

Output Early

The **Early** control adjusts the level of early reflections within the reverb output signal.

Output Late

The **Late** control adjusts the level of late reflections within the reverb output signal.

Pattern Delay

This effect features 2 delay lines - the first is a multitap delay sets the 'pattern' and the second is a simple stereo delay with feedback. The length of the 2 delay lines is distinct although they are both set as multiples of the host tempo using the **Step Length** setting.

• **Pattern delay line (multitap, no feedback)**: maximum length = 1/4 note * 16, min length = 1/64T * 16

• **Repeat/Regen delay line (simple stereo, feedback)**: max. length 1/4 note * 16 / minimum 1/64T * 1

The Pattern delay line cannot be fed back on itself without engaging the Repeat/Regen delay line.

Controls for each of 4 taps

Steps (1-16)

Each of the 4 taps can be placed on 1 of 16 'sequence' steps which, combined with the Step Length and Repeat parameters, define the 'pattern' in the Pattern Delay. This pattern comprises the first delay line.

Click the number of each tap to enable/disable it in the sequence.

Level / Pan / Pitch / Res

These controls set the volume, stereo pan position and filter settings for each of the 4 taps.

Blend

The Blend control crossfades between the following signals for each of the 4 delay taps:

Unfiltered - Low-pass - Band-pass - High-pass - Unfiltered

Try using modulation to sweep this control between the various filter types while delay taps are active and feeding back with the Repeat/Regen controls.

Global controls

Repeat

This parameter defines the length of the Repeat/Regen delay line which exists to feed the output back into the Pattern delay line's multiple taps. It is set in steps so its length is dependent on the **Step Length** parameter setting.

• With a **Repeat** setting of *8* and a **Step Length** of *1/16*, the pattern is fed back on itself every half-bar.

• With a **Repeat** setting of *8* and a **Step Length** of *1/8*, the pattern is fed back on itself every bar.

• With a **Repeat** setting of *8* and a **Step Length** of *1/32*, the pattern is fed back on itself every quarter-bar.

Deactivate the **Repeat** button to entirely disable the Repeat/Regen delay line.

Regen

This parameter sets the amount of feedback for the Repeat/Regen delay line.

RG Mix (Regen Mix)

The **RG Mix** parameter controls the signal mix which is routed to the feedback circuit - it mixes between the Pattern Delay line and the Repeat/Regen delay line signals.

HP / LP

These High-pass (**HP**) and Low-pass (**LP**) filters exist within the Repeat/Regen delay line's feedback path and are be used to filter out low and high frequencies in the feedback signal.

Step Length

The **Step Length** parameter defines the length of each Step, in terms of a BPM-division of the master tempo, used in both delay lines.

TinCanVerb

This effect is a recreation of a low-end room reverb unit, perfect for emulating 'cheap and nasty' onboard synth FX. Use the FX-Verb device for higher quality reverb.

Size

The **Size** control adjusts the size of the virtual reverberation room. Smaller rooms offer subtle ambience, while large rooms result in a more 'cavernous' and reflective effect.

Decay time

This control adjusts the **Decay time** of the reverb effect.

Use shorter settings for subtle small room effects and longer times for special effects.

Damp

Increasing the **Damp** control results in less high frequencies in the effected signal, leading to a darker reverb sound. At least some damping is essential to avoid overly tinny and fatiguing results.

Pinch / Squeeze

The **Pinch** and **Squeeze** controls manipulate the shape of the virtual reverb room, leading to a variety of resulting effects. They make the reverb sound a lot more artificial and are useful for special effects.

Freeze

As the name suggests, this button 'freezes' the current reverb buffer and loops it indefinitely until the button is disabled. Modulating the **Freeze** control is useful for dubby special effects.

Freezer

This is a granular buffer-looping/freezing effect that is useful for glitchy and stuttery sounds.

Sync

With the **Sync** button activated, the **Length** and **Grain Size** parameters are set in tempo-based values relative to Cypher2's current tempo. Possible values range from 64th note to 2 bars, including dotted and triplet variations. With the button deactivated, these parameters are set in seconds.

Gate / Length

Once the **Gate** control is turned to the *On* position the Freezer effect starts recording audio from the input into a buffer.

The size of this buffer is dictated by the **Length** control (1-16 beats in **Sync** mode, up to 2 seconds with Sync deactivated).

The loop buffer is filled until the end of the time period defined by the **Length** parameter, after which subsequent incoming audio is ignored until the buffer is re-gated (**Gate** control set to *Off* and *On* again).

Grain Size

The audio in the loop buffer is divided up into slices, the size of which is dictated by the **Grain Size** parameter. This can be set in seconds or BPM units depending on the setting of the **Sync** button.

Smooth

The **Smooth** control introduces crossfading to each grain, relative to the size of the grain. At a setting of 100% the crossfading occurs over the entire length of the grain. At a setting of 0, no crossfading occurs.

Speed / Jump Manual / Jump Rand

Once the buffer has been filled, the Freezer loops the first grain – it 'freezes' the grain – while the **Speed**, **Jump Rand** and **Jump Manual** parameters are set to 0.

Increasing the **Speed** control plays through the grains sequentially starting with a grain between 0 and 16, set using the **Jump Manual** parameter, as long as **Jump Rand** = 0. A setting of 100% is normal speed.

Increasing the **Jump Rand** parameter results in the Freezer jumping to random grains instead of playing through them sequentially.

Scratch

The **Scratch** parameter scales the pitch of the loop just like a record on a turntable - the loop can be played forwards and backwards and everywhere in between.

7.4 Modulation FX

Phaser

The Phaser uses phase cancelling techniques (with the use of all-pass filtering) to create a series of peaks across the frequency spectrum. When these peaks are moved over time, a psychedelic sweeping effect is created.

Mode

The **Mode** control selects between a number of phaser responses. 4, 6, 8 and 12 stage phaser types are available, with positive or negative feedback. The number of stages refers to the number of all-pass filters within the algorithm.

Pitch

The **Pitch** control adjusts the centre frequency of the all-pass filters used in the Phaser algorithm.

Resonance

The **Resonance** control adjusts the amount of resonance (feedback) in the all-pass filters.

Sync

When the **Sync** parameter is set to *BPM*, the **Rate** control is set in tempo-based values relative to Cypher2's current tempo. With Sync deactivated, the Rate control is set in seconds.

Rate

The **Rate** control adjusts the speed of the internal LFO. In **Sync** mode, possible values range from 64th note to 2 bars, including dotted and triplet variations. With Sync deactivated, the available range is 31ms to 4 seconds.

Depth

The **Depth** control adjusts the amount of Pitch modulation from the Phaser device's internal LFO that modulates the **Pitch** parameter for classic phaser effect sounds.

Phase

The **Phase** control adjusts the phase between the dry signal and the all-pass filtered signal.

Flanger

The Flanger effect is a short modulated delay line with feedback to the input. It is used for a sense of movement and for psychedelic effects from the subtle to the extreme. It features a built-in sine LFO for modulating the flanging delay line.

Freq (Frequency)

The **Freq** control affects the speed of sine LFO modulation of the Flanger's delay time.

Depth

The **Depth** control adjusts the amount of modulation of the delay time.

Length

This control introduces an additional fixed delay time to the Flanger's delay line. It is a very short delay, ranging from 0 ms to 15 ms.

Feedback

This control adjusts the amount of the flanged signal that is fed back into the input.

Higher **Feedback** settings result in a more pronounced flanging effect. Settings over 50% lead to extreme comb filter type effects.

Spread

This control varies the panning of the left and right channel processed signals.

Phase

The **Phase** control offsets the phase of the internal LFOs for the left and right channels.

Invert

By default (with the **Invert** button deactivated), the flanged signal is in positive phase with the input signal.

Activating the **Invert** button inverts the flanged signal's phase in relation to the input signal.

A positive phase setting tends to lead to a more obvious flanging effect.

Chorus

The Chorus effect is a modulation effect that is pitch-based. It is used for thickening up sounds.

Rate

The **Rate** control adjusts the speed of pitch modulation by the built-in sine LFO.

Depth

The **Depth** control adjusts the amount of modulation away from the input signal's original pitch.

Spread

This control varies the panning of the left and right pitch-modulated signals.

Autofilter

This device features a state-variable multimode filter with a response of 12 dB/oct, capable of self-oscillation and featuring input and output drive stages.

It has a built-in envelope follower to modulate the filter's cutoff frequency relative to the input's amplitude. This function is modelled on an analogue full-wave rectified envelope follower. Additionally, the filter frequency can be modulated at audio rate by the input signal.

Mode

4 filter Mode settings are available, each offering different filtering functions relative to the cutoff frequency.

Low (Low-pass)

Low-pass mode allows through only frequencies below the cutoff frequency.

High (High-pass)

High-pass mode allows through only frequencies above the cutoff frequency.

Band (Band-pass)

Band-pass mode allows through only a band of frequencies around the cutoff frequency.

Notch

This is the opposite of a band pass - it allows through all other frequencies except a band of frequencies around the cutoff frequency.

Modulating a notch filter can give phaser-like results.

Drive

The **Drive** control sets the amount of signal gain before the filtering stage. As well as increasing the gain, using more drive results in a rich and complex interaction with the filter's resonance due to modelled non-linearities within it.

Out Drv (Out Drive)

The **Out Drive** control adjusts the gain of an OTA-type non-linear amp function at the output, in order to boost and colour the filtered signal.

Cutoff

The **Cutoff** control adjusts the cutoff frequency of the filter. It is measured in semitones.

Res (Resonance)

This control adjusts the resonance of the filter, accentuating the frequencies around the cutoff frequency set by the **Pitch** control.

High **Resonance** settings cause the filter to self-oscillate.

Attack

This control changes how quickly the envelope follower section responds to transients in the input signal. Longer **Attack** times mean that the filter takes a longer time to respond to amplitude changes.

Release

The **Release** control changes how quickly the envelope follower causes the filter to return to its original position when the input signal decreases after a transient.

Depth

The **Depth** control adjusts the amount of modulation of the filter's cutoff frequency by the built-in envelope follower.

FM

The **FM** control sets the amount of audio-rate modulation of the cutoff frequency by the input.

Amber Chorus

Amber Chorus provides a range of chorus algorithms derived from classic bucket-brigade delay (BBD) chorus circuits found in string-synthesizers and stomp boxes. These algorithms impart a psychedelic and thickening effect to the source signal and results in a 'blended' ensemble effect with a very different character to the regular Chorus effect.

Mode

6 different models are available using the **Mode** control with each providing their own unique sonic character:

- 1975
- 1977
- 1981
- 1984
- 1985
- 1986

Speed

This control adjusts the rate of Pitch modulation.

Spread

This control varies the panning of the left and right pitch-modulated signals.

Bright

The algorithms in Amber Chorus were derived from classic ensemble string synth chorus circuits. Such devices usually integrated additional fixed filtering to shape the tone of the output to be more string-like. Activating the Bright button bypasses these filters. With the button deactivated, the filters are applied and a darker sound is created at the output.

Amber Formants

Amber Formants is a 4-band formant filter consisting of 4 bandpass filters. It is useful for vocal-style filtering and for imparting string-like characteristics to sounds.

The formant filter is capable of self-oscillation – use the final mixer/amplifier controls to tame the sound so that it does not distort at higher resonance settings.

Freq (Frequency)
Gain

The formant filter comprises 4 band-pass filters, each with their own **Freq** (frequency) and **Gain** controls.

Resonance

The resonance of all bands is controlled by a single **Resonance** parameter

Scale

The **Scale** control adjusts the frequency of all 4 bands simultaneously (relative to their individual **Freq** settings).

Notch

The **Notch** button switches all bands to notch filter mode – in this mode, the **Gain** controls are inactive.

RingMod

Ring modulators are used for radical timbral shifts and experimental effects. It multiplies the input signal with an internal oscillator, creating interesting sidebands and inharmonic timbral changes in the output signal.

Mode

The **Mode** control adjusts the waveshape of the internal oscillator which is multiplied with the audio input to the effect. *Sine*, *Triangle*, *Saw*, *Square* and *Parabolic* oscillator shapes are available, as well as *White* or *Pink* noise.

Pitch

The **Pitch** control tunes the internal oscillator within a range of 1 octave.

Drive

The **Drive** control introduces an adjustable amount of distortion on the input signal. Overdriving the signal in this way changes the waveshape of the input, leading to further variations in the resulting effect.

Nonlinear RingMod

This device provides an alternative ring modulator circuit. Again, the input signal is multiplied with an internal oscillator to create complex sidebands and dissonant effects.

Drive

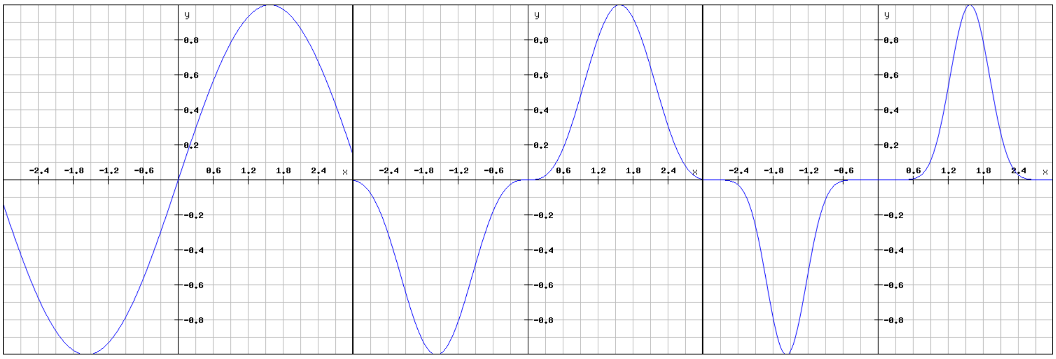
The **Drive** control adjusts the amount of overdrive on the input signal. Higher amounts of Drive affect the waveshape and harmonics of the input signal and lead to further variation in the timbre of the output.

Pitch

The **Pitch** control tunes the internal oscillator by +/- 6 octaves.

Mode

The **Mode** control adjusts the waveshape of the internal oscillator between *Sine*, *Triangle*, *Sine3* and *Sine9* shapes. The latter shapes represent $\sin(x)^3$ and $\sin(x)^9$ functions respectively. The following graphs show how their shapes differ from a regular sine shape.



Sine Sine3: $\sin(x)^3$ Sine9: $\sin(x)^9$

Tone

The **Tone** control introduces and adjusts a low-pass filter function to the output signal. Increase the control to increase the amount of filtering.

PhaseMod Array

The PhaseMod Array device provides 4 sine oscillators whose phase is modulated at audio-rate by the incoming signal. It is designed to provide audio-rate FM effects. When using higher amounts of the **Spread** and **Depth** controls, the device often produces best results with a monophonic synth source.

Pitch

The **Pitch** control adjusts the base tuning of 4 sine wave oscillators.

Stack / Spread

The **Stack** control dictates how many of the 4 sine oscillators are heard while the **Spread** control spreads the pitch of the 4 sine oscillators apart according to the **Mode** setting. These functions behave in a very similar way to the **Stack** and **Detune** functions in Cypher2's oscillator.

Width

The **Width** control spreads the 4 sine oscillators apart in the stereo field.

Mode

The **Mode** control changes the way the **Spread** control affects the tuning of each of the 4 sine oscillators.

The following table shows the deviation from the base tuning (set by the **Pitch** control) of each sine introduced by the **Stack** control with the **Spread** control at 100%:

	Mode A	Mode B	Mode C
Stacked sine 1	+0	+0	+0
Stacked sine 2	-12	-11	+11
Stacked sine 3	+12	+12	-12
Stacked sine 4	-5	+7	-4

Depth

The **Depth** control sets the amount of audio-rate phase modulation applied to the 4 sine oscillators from the audio input signal.

FreqShift (Frequency Shifter)

The FreqShift effect changes the pitch of a signal without preserving the harmonic information, resulting in very alien, abstract and clangorous timbres.

Pitch

The **Pitch** control adjusts the amount of frequency shifting by +/- 3 octaves.

Gain

The **Gain** control adjusts the level of the output signal.

7.5 Distortion FX

Bitcrusher

The Bitcrusher effect provides a type of digital distortion that occurs when the sample-rate and bit-depth of the audio is reduced. It simulates the sound of early samplers, useful for underground hip-hop and other 'lo-fi' styles.

Bits

The **Bits** control reduces the bit depth from a maximum of 16 bits to a minimum of 1 bit, which is effectively digital noise. The digital noise generated by the bit-reduction process is referred to as quantisation noise.

Frequency

The **Frequency** control adjusts the sample rate frequency of the audio processed by the effect and ranges from a maximum of 100 kHz to a minimum of 1 Hz.

Lower sample rates result in an aliasing effect on the processed audio.

Drive

The **Drive** control adjusts the amount of drive in an OTA-type distortion stage after the crossover filters. This allows gain and colour to be added to the signal before it is processed by the **Bit** and **Freq** processes.

LPF (Low-pass filter)
HPF (High-pass filter)

These controls provide high-pass and low-pass for isolating a part of the signal before the sample-rate and bit-depth reduction stages.

These filters act as crossover filters – the active frequency band is processed by the distortion circuit with its level adjustable via the **Dirty** control. The frequencies that are filtered out before the distortion stage are accessible via the **Clean** control.

Dirty & Clean

The **Dirty** control sets the amount of post-distortion signal that is heard at the output.

The **Clean** control sets the amount of the clean signal, which is comprised of the signals filtered out by **LPF** and **HPF** filters.

Please note that these are not 'wet' and 'dry' controls – use the standard **Mix** control at the upper-right of the effect interface in order to mix between the pre- and post-effect signals.

Tone

The **Tone** control provides a simple -6 dB/oct low-pass filter for the **Dirty** signal after the bitcrushing process. It rolls off unwanted high frequencies that may have been generated in the signal as part of the distortion effect.

Drive

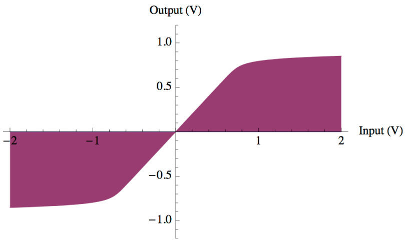
Drive is a versatile DCAM-modelled overdrive/distortion effect.

Drive

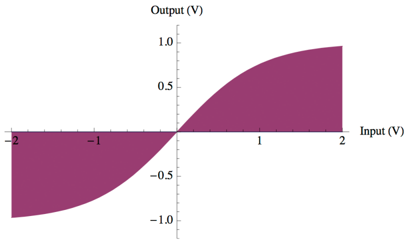
The **Drive** control sets the amount of distortion that takes place.

Mode

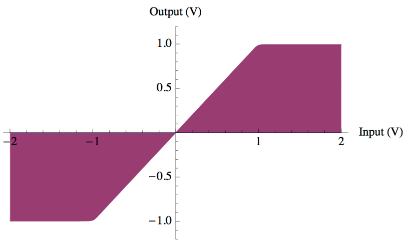
The **Mode** drop-down menu selects between 4 distortion types:



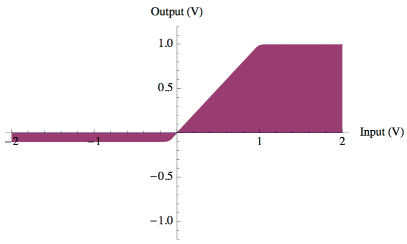
- *Diode*



- *OTA*



- *OpAmp*



- *HalfRect (half-rectified)*

LPF (Low-pass filter)
HPF (High-pass filter)

These controls provide high-pass and low-pass filters before the distortion stage to shape the tonal characteristics of the signal going into the drive circuit.

For example, it may be useful to distort the high end of a sound while leaving the low end unchanged.

These filters act as crossover filters – the active frequency band is processed by the distortion circuit with its level adjustable via the **Dirty** control. The frequencies that are filtered out before the distortion stage are accessible via the **Clean** control.

Dirty & Clean

The **Dirty** control sets the amount of post-distortion signal that is heard at the output.

The **Clean** control sets the amount of the clean signal, which is comprised of the signals filtered out by **LPF** and **HPF** filters.

Please note that these are not 'wet' and 'dry' controls – use the standard **Mix** control at the upper-right of the effect interface in order to mix between the pre- and post-effect signals.

Tone

The **Tone** control provides a simple -6 dB/oct low-pass filter for the processed signal after the distortion stage. It rolls off unwanted high frequencies that may have been generated in the signal as part of the distortion effect.

DirtyDAC

This effect emulates the behaviour of an old Digital-to-Analogue Converter with low-fidelity characteristics. It offers an alternative processing algorithm to the Bitcrusher device and imparts a similar type of effect, although the character is very different.

Frequency

The **Frequency** control sets the sample-rate frequency of the DAC model. It can be adjusted between 1kHz (fully anti-clockwise) and 10kHz. Lower settings lead to more pronounced aliasing effects.

Anti-alias

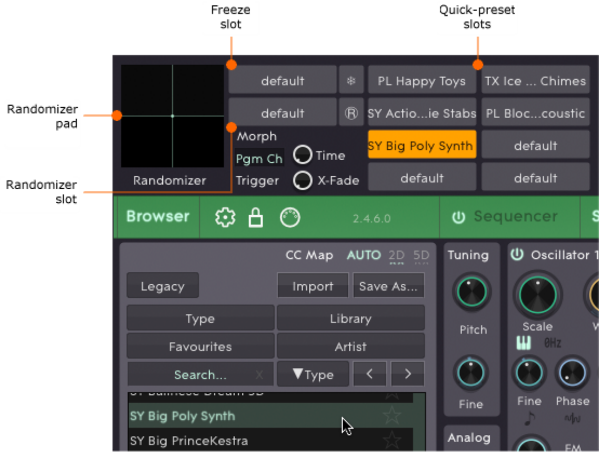
The **Anti-alias** control adjusts the amount of anti-aliasing filtering for the output signal. Turning up the control reduces the cutoff frequency of a low-pass filter, reducing audible sidebands and aliasing effects caused by the sample-rate reduction process.

Bits

The **Bits** control sets the bit-depth of the modelled DAC - it can be set between 6 and 16 bits. Lower values introduce more noise into the signal and reduce the dynamic range.

8 Quick-presets, Morphing and Randomizing

Quick-presets overview



• Loading a new preset to the current slot

The upper-left part of the Cypher2 interface contains the Quick-preset, Morph, Freeze and Randomizer controls.

These functions involve a total of 10 slots which each hold an entire preset (state of the Cypher2 instrument). Quick-preset slots can be considered as multiple 'edit buffers' - any editing of Cypher2 always applies to the currently selected Quick-preset slot. 1 of these 10 slots is always active as the *currently selected Quick-preset slot* and highlighted in *yellow*.

• When a preset is loaded from the Browser, it is loaded into the currently selected Quick-preset slot.

Main 8 Quick-preset slots

The 8 main Quick-preset slots can each be used to hold a full Cypher2 preset. Each slot can either be filled in either of the following ways:

• Select a Quick-preset slot by clicking it and load a preset from Cypher2's Browser - the preset is loaded to the Quick-preset slot, leaving other slots unaffected.

• Copy contents between Quick-preset slots using the Quick-preset context menu (right-click on the slot)

Freeze slot

The Freeze slot and its accompanying **Freeze** ❄ button is used for creating new sounds using the morphing functions with sounds within the 8 main Quick-preset slots.

It also features a **History** function which effectively provides a multi-level undo function for its contents.

Randomizer slot

The Randomizer slot, **Randomizer** ® button and Randomizer pad are used for creating new sounds through randomization. Like the Freeze slot, the Randomizer slot also features a **History** function to return to previous states.

Using Quick-presets

Quick-presets are intended to be used for a variety of functions:

Fast switching between 8 presets for live use scenarios

Quick-presets allow extremely fast switching between sounds. If switching between Quick-presets with the same Effects page contents (or when using the Lock function to prevent them being changed), the transition to the new preset occurs extremely quickly compared to loading presets from disk using the Browser (under 1ms vs. 10+ ms).

• Quick-preset slots can be selected using MIDI messages (a range of notes, program changes or MIDI CC #0).

• Try also creating copies of a sound to other slots using the Quick-preset context menu. Then program variations which can subsequently be recalled in real time via MIDI.

Exploring variations of sounds using the morphing and Freeze functions

With the morphing and Freeze functions, it is possible to morph between 2 Quick-preset slots, creating new sounds with Freeze operations during mid-morph states. The results can be subtle or radical depending on whether similar or wildly different presets are used.

Exploring variations of sounds using the Randomizer functions

The Randomizer provides a different way of exploring new sounds - it creates 4 random seeded variations which can then be continuously explored using the Randomizer Pad.

Selecting Quick-preset slots with the mouse/trackpad or computer keyboard

• Selecting another slot by clicking

Click any Quick-preset slot to switch to it, selecting it as the current slot. If the **Time** control is set to *0*, the preset within the slot is recalled immediately.

If the **Time** control is increased, a morph occurs for the duration of the Time setting. During this period, parameters morph to the values in the new preset. Morphing is discussed in detail in the next section.

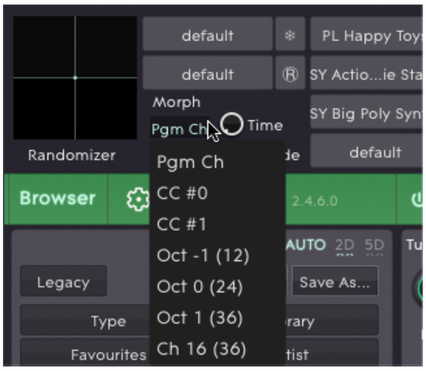
To select a Quick-preset slot with the mouse/trackpad immediately regardless of the **Time** control setting, select it while holding down the ALT key.

Quick-preset slots can also be selected using the following keyboard cursor shortcuts (in DAW/host applications which do not intercept keyboard input):

←	Previous Quick-preset slot
→	Next Quick-preset slot

Selecting Quick-presets via MIDI

Quick-presets can also be selected using the following MIDI messages, selectable using the **Trig** menu.



• *Pgm Ch* (Program change)

MIDI Program change messages 1 to 10.

• *CC #0* or *#1* (MIDI continuous controller #0 or #1)

MIDI CC#0 or #1 is used to address Quick-preset slots 1-10 proportionally over the CC value range (0-127).

• *Oct -1 (12)* / *Oct 0 (24)* / *Oct 1 (36)*

Notes C to A in 1 of 3 octaves, starting from MIDI note number 12, 24 or 36, on MIDI channel 1.

• *Ch 16 (36)*

Notes C1 to A1 starting from MIDI note number 36, on MIDI channel 16.

Quick-preset context menu

Right-click on any Quick-preset slot to display the Quick-preset context menu.

Copy

This function copies the Quick-preset slot's contents and stores them in the clipboard.

Paste

This function overwrites the Quick-preset slot's contents with the contents of the clipboard.

Reset

This function clears the Quick-preset slot and replaces its contents with the Init patch.

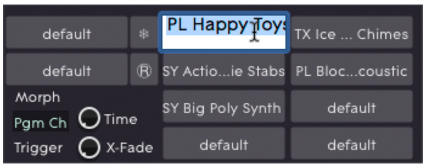
Clone to all

This function copies the contents of the Quick-preset slot to all 7 other main Quick-preset slots (the Freeze and Randomizer slots are not affected).

Reset All

This function clears the contents of all 8 main Quick-preset slots (the Freeze and Randomizer slots are not affected).

Renaming the preset within a Quick-preset slot



Double-click any Quick-preset slot (including the Freeze and Randomizer slots) to enter a new name for the preset within it. Note that this action does not save the preset - select the slot and use the **Save As** function in the Browser or Settings menu in order to save the slot's contents as a preset.

Saving/recalling Quick-preset setups

Because they involve compiling multiple single Cypher2 presets, Quick-preset setups exist *outside Cypher2's preset format* - any time a preset is saved using the Save As function in Cypher2's Browser or Settings menu, only the **currently active** Quick-preset slot's contents are saved.

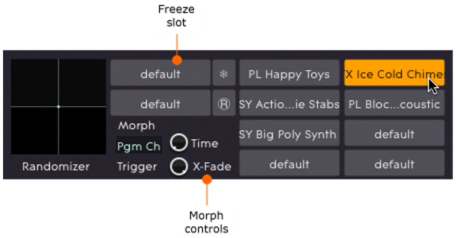
However, Quick-preset setups can be saved and recalled in the following situations:

- Within the host project file when Cypher2 plugin instances are used in a host/DAW.
- Within host/DAW preset files - such as ***.FXP** in Cubase/Nuendo or ***.AUpreset** in Logic for example)

8.1 Morphing and Freezing

Morphing

Timed morphing using the Time (Morph Time) control



When a Quick-preset slot is selected, either by clicking the slot or using a MIDI message specified via the **Trigger** menu, the transition from the previous state to the new slot's contained preset - called the *morph destination* - occurs over a period defined by the **Morph Time** control.

During this period, Cypher2's parameters are moved continuously in real time to new positions as dictated by the new preset.

If Cypher2 is played via MIDI (or with a locked Sequencer/Arpeggiator) during this period, the effect of the parameters moving to new positions is heard.

Interrupting a timed morph

• If any controls are adjusted while a timed morph is in progress, they are excluded from the morph (which otherwise continues as normal) and their final positions are added to the state stored in the *morph destination* Quick-preset slot.

• If a new preset is loaded from the Browser while a timed morph is in progress, the morph process is cancelled and the new preset is loaded to the *morph destination* Quick-preset slot.

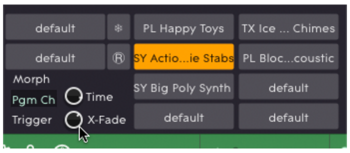
Manual morphing with the X-Fade control

The main 8 Quick-preset slots are arranged in 2 adjacent banks of 4.

If a Quick-preset slot in the left bank is currently selected and a slot on the right is subsequently selected, the **X-Fade** control is turned fully to the right at the end of the morph (and vice versa). It is always fully turned towards the currently selected Quick-preset slot and is visible *only* when using the 8 main Quick-preset slots - it cannot be used when either the Freeze or Randomizer slots are selected.

The **X-Fade** control can be used to manually morph towards most settings of the Quick-preset slot immediately to the right or left of the currently selected slot. This manual morphing process excludes both of the following:

• TransMod slot assignments (but not modulation amounts within TransMod slots)
• Effects page contents



The selected Quick-preset slot *does not change* - the original slot remains selected even with the **X-Fade** control turned fully away from it.

While a manual morph is in progress (when the **X-Fade** control is at any setting except fully turned towards the currently selected Quick-preset slot):

• Using the **Freeze** ❄ button adds the current state of Cypher2 to the Freeze slot and the Freeze slot is selected as the current Quick-preset slot

• Using the **Save As** function adds the current state of Cypher2 to the Freeze slot, the Freeze slot is selected as the current Quick-preset slot and its contents are saved as a preset

• Moving any Cypher2 parameters results in the new state being added to the Freeze slot and the Freeze slot is selected as the active Quick-preset slot

• Loading a new preset from the Browser results in the preset being loaded to the currently selected Quick-preset slot and the **X-Fade** control is fully turned towards it

See below for more details of the Freeze slot and its operation.

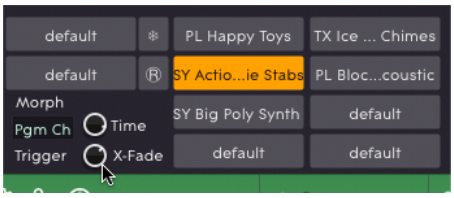
The **X-Fade** control is not available while a timed morph is already in progress.

Selecting Quick-preset slots without morphing

To switch to a preset without morphing (and without reducing the **Morph Time** control to 0), click the Quick-preset slot while holding down the ALT key.

Using the Freeze slot

The Freeze slot (which also features a corresponding **Freeze** ❄ button to its right) is a special Quick-preset slot which stores 'frozen' mid-morph states in order to create new sounds. The most immediate way to use the Freeze slot is as follows:



Create a manual **X-Fade** towards the adjacent Quick-preset as shown previously and then click the **Freeze** ❄ button.

The current state of Cypher2 is 'frozen', stored in the Freeze slot and named *(Freeze 01)*. The **X-Fade** control is not accessible with the Freeze slot selected.

The Freeze slot is filled with the current state of Cypher2 and selected as the active Quick-preset slot in all the following situations:

• The **Freeze** ❄ button is clicked while a manual **X-Fade** morph is in progress (when the **X-Fade** control is at any setting except fully turned towards the currently selected Quick-preset slot) - as shown above

• The **Freeze** ❄ button is clicked while a timed morph towards a Quick-preset is in progress

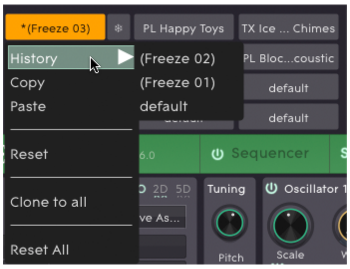
• The **Save As** function is used while an **X-Fade** morph is active - the frozen state is also the state which is saved

• A Cypher2 parameter is adjusted while a **X-Fade** morph is in progress - the current state of Cypher2 is saved to the Freeze slot along with any manually adjusted controls' final positions

Selecting the Freeze slot

The Freeze slot can be selected at any time, like any other Quick-preset slot, in order to recall its current contents and without creating a new Freeze state.

While the Freeze slot is selected, any new preset loaded from the Browser is loaded into it.



Freeze slot History

When the Freeze slot is filled with a new state, the previous state is stored in the Freeze slot History. Right-click on the Freeze buffer slot to to display a list of sequentially stored Freeze slot states - click any state to recall it.

Using Freeze slot contents

When the Freeze slot is selected, its contents can be saved as a preset in the same way as any other Quick-preset slot.

It can also be useful to copy Freeze slot contents to other Quick-preset slots with the **Copy**/**Paste** funtions in the Quick-preset context menu in order to compile variations or to perform further editing.

Using Locks during morphing

Please note that while transitions of continuous parameters are smoothly interpolated during morphing operations, switched controls and differing TransMod slot assignments may produce abrupt changes or clicks. It is also possible to encounter clicks when the contents of the Effects page are morphed.

Using Locks on certain parts of Cypher2 such as the Effects page and TransMod slot setups can be very useful during morphing operations as some causes of abrupt changes and clicks can be avoided - these 2 elements are always locked during manual **X-Fade** morphing as mentioned above.

When simply using the morphing functions to experiment with creating new sounds, clicks produced during the process may not be a concern.

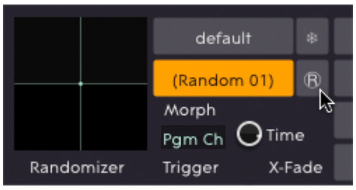
8.2 Randomizer

Using the Randomizer slot

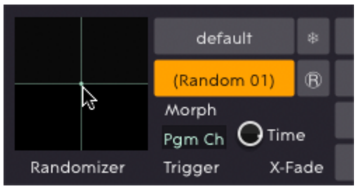
The Randomizer slot (which also features a corresponding **Randomizer** ® button to its right) is another special Quick-preset slot which is used in combination with the Randomizer Pad for creating new sounds.

To use the Randomizer:

1



Either click the **Randomizer** ® button...



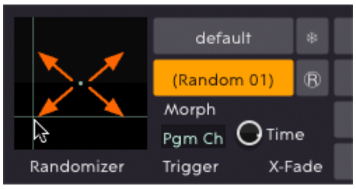
... or click and keep the mouse/trackpad button held down on the Randomizer Pad

At this point:

• The current state of Cypher2 is copied to the Randomizer slot and given a new *(Random)* name, in a similar way to clicking the **Freeze** ❄ button to populate the Freeze slot. This creates an 'undo' point to return to the original state using the History menu if desired after creating a random variation.

• The Randomizer slot is also selected as the active Quick-preset slot.

2



Click and hold on the Randomizer Pad if this was not already done in step 1.

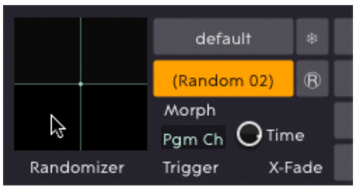
Then, keeping the mouse/trackpad button held down, drag within the Randomizer pad to explore random variations.

Cypher2 can, of course, be played during this process.

As soon as the Randomizer pad is clicked, the 4 corners of the pad are re-seeded with new random parameter values, with the original state located at the centre.

Dragging towards each corner gradually morphs towards its random values from the original state.

3



Release the mouse/trackpad button at the desired point to store the live state as the new contents of the Randomizer slot, adding the previous contents of the slot to its slot History.

Selecting the Randomizer slot

Like the Freeze slot and any other Quick-preset slot, the Randomizer slot can be selected at any time in order to recall its current contents and without creating a new Randomized variation.

While the Randomizer slot is selected, any new preset loaded from the Browser is loaded into it.

Randomizer slot History

When the Randomizer slot is filled with a new state, the previous state is stored in the Randomizer slot History. Right-click on the Randomzer slot to to display the History menu - a list of sequentially stored Randomizer states - and click any state to recall it.

Using Randomizer slot contents

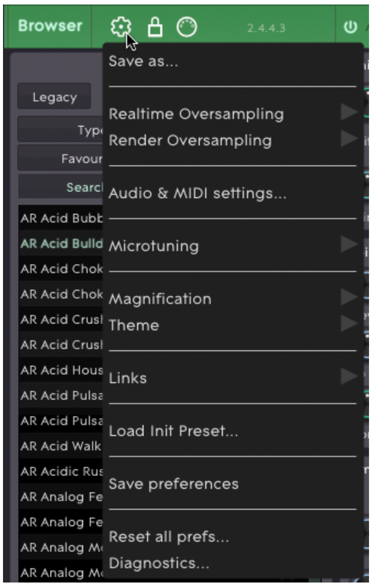
• When the Randomizer slot is selected, its contents can be saved as a preset in the same way as any other Quick-preset slot - click the **Save As** function in the Navigation bar or Settings menu to proceed.

• It can also be useful to save Randomizer slot contents to other Quick-preset slots with the **Copy**/ **Paste** functions in the Quick-preset context menu in order to compile variations or to perform further editing.

Using Locks during Randomizer operations

Locks can be very useful during Randomizer operations - for example, it may be convenient to keep the Pitch control, TransMod slot assignments or Effects page contents locked.

9 Settings menu



Click the **Settings** ⚙ button to show the Settings menu which adjusts various global settings that are not stored within presets.

Save as

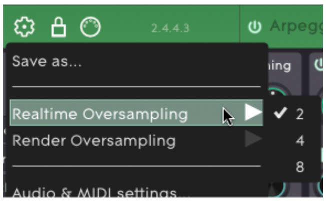
This function duplicates the **Save as** button in the Browser.

Reset all prefs...

The **Reset all prefs** function resets the preference settings to their factory default values.

Realtime and Render Oversampling

These settings relate to the oversampling in the synth engine. Higher oversampling sounds better but uses more CPU!



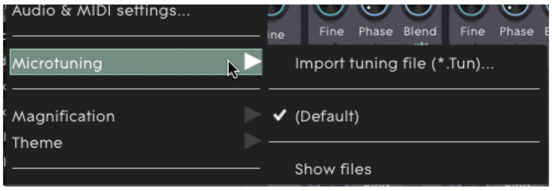
The **Realtime Oversampling** setting relates to the oversampling used for normal realtime operation. Set this as high as the system's CPU (and the number of voices) allows.

The **Render Oversampling** setting relates to 'offline' mixdowns and 'freeze' operations in hosts that provide this feature. This setting can be set to very high levels regardless of the system CPU for extremely high sound quality. However, higher settings result in longer render times for offline mixdown/freeze operations.

Audio & MIDI settings...

This function is only available in the standalone version and allows the desired **Sample rate**, **Buffer size** to be specified, as well as the required **MIDI Inputs** and **Audio Output**. Please note that the **Input device**, **Audio Inputs** and **MIDI Outputs** sections in this dialog are redundant in Cypher2. Click the **Done** button to return to the Cypher2 interface.

Microtuning

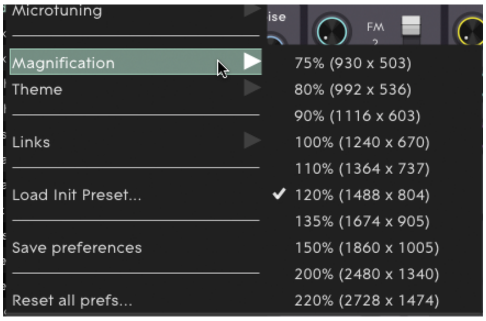


The **Microtuning** sub-menu allows several operations for using microtonal tuning files (.Tun files).

The **Import tuning files** function imports .Tun files to Cypher2's internal database of tuning files. The menu of scales allows selection from the available imported files.

To return to regular 12-tone scale operation, select *(Default)* in the menu. The **Show files** function opens the folder on disk in which tuning files are stored.

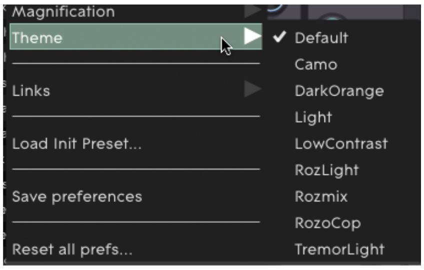
Note that Cypher2's internal handling of pitch is always in an equal-temperament 12-tone scaling - it can be considered in a similar way to the 1 volt per octave system used in modular synthesizers. Microtuning is applied during the point at which MIDI input is converted into 'pitch volts' - control signals that dictate keytracking values which set the osc pitch when being assigned to a voice.



Any additional adjustment and modulation within Cypher2 of pitch and other keytracked parameters is conducted within its normal equal-temperament 12-tone scaling.

Magnification

Cypher2 features a vector-drawn interface which can be scaled to a variety of sizes, selectable in the **Magnification** sub-menu.



Theme

This sub-menu allows toggling between 2 available interface themes for the Cypher2 interface - *Dark Theme* and *Light Theme*.

Links

This sub-menu provides various web-links which open in the system's default browser.

Load init preset...

This function loads the IN Init preset to the current Quick-preset slot after a confirmation dialog.

Diagnostics...

This function reports CPU capabilities and performance.

Our tech support staff may ask to use this function during troubleshooting - it can be otherwise ignored.

10 Cypher2's Oscillators in detail

This section features extended discussions around the specialized audio-rate modulation and other osc modulation functions within Cypher2:

Frequency Modulation and Wave Modulation

• Audio-rate frequency modulation is available in the osc section (with Osc1's **FM 3** and **FM 2** controls and Osc2's **FM 3** control) and in the Filter sections (using the **FM 3** controls).

• Audio-rate waveform modulation is provided by Osc3's **WM 2** control.

Sample & Hold, Sync and Beat Detune

• Cypher's audio-rate S+H functions provide low-fidelity FM-esque timbres, simulating low-resolution digital technology. The **Blend** parameter at the minimum position provides the S+H signal for each osc.

• The Sync section describes Cypher's variable-depth osc-sync functions, using the **Sync 1** controls on Osc2 and Osc3.

• The Beat Detune section describes how each osc's individual **Fine** controls allow consistent 'beating rates' across the keyboard range.

10.1 Frequency and Wave Modulation

Introduction to audio-rate modulation

While subtractive synthesis – using a filter to subtract portions of an oscillator's frequency range – is relatively easy to understand, FM and other audio-rate modulation tends to perplex many musicians.

Conventional subtractive synthesis is great for many types of mostly abstract sounds but it isn't usually sophisticated enough to produce more complex, 'realistic' timbres such as those produced by acoustic instruments. This is because most real-world timbres involve very complex harmonic variation – harmonics change at very high speeds and in more complex ways than those made possible by subtractive filtering.

Basics of audio-rate modulation

Most modulation in subtractive synthesizers takes the form of control-rate functions such as LFOs and envelopes, which do not produce fast enough modulation for these complex timbres.

Audio-rate modulation – modulation at frequencies of audible sounds – is capable of producing more complex tones. FM is one of the most common types of audio-rate modulation, and can be very effective at emulating many types of acoustic sounds, particularly when using envelopes and other control-rate modulation to dictate the amount of FM applied over time.

FM can be very good at approximating the sound of acoustic plucked and hammered instruments in particular. It can also be used for producing new types of abstract sounds that are not possible with conventional subtractive synthesis.

Audio-rate modulation implementations

FM (frequency modulation) is perhaps the most well-known type of audio-rate modulation. While it is common to associate FM with digital synths such as the Yamaha DX and TX series, FM capabilities can actually be found on many analogue synthesizers. Many classic poly- and monosynths feature FM capabilities, with one osc typically modulating another osc or the filter. Some examples include the Roland Jupiter series, SCI Prophet 5, Minimoog and ARP Odyssey. Most patchable modular and semi-modular synthesizers also allow this kind of functionality.

Digital synthesizers have a huge advantage over analogue synths when it comes to FM: it is easy to input exact values and frequency ratios while tuning and keyboard tracking tends to be stable and precise - something that is much harder to achieve in analogue systems.

Despite these drawbacks, the sound of analogue FM is rich and complex. With its meticulously modelled oscillators, Cypher2 attempts to combine the sound of analogue with the convenience and stability of a digital polyphonic software instrument.

Frequency modulation

When the frequency of one oscillator is modulated with that of another, the modulator oscillator is essentially acting as a very fast vibrato on the modulated osc's pitch.

An LFO is an oscillator running at low speeds, which produces a vibrato effect when it modulates pitch.

Modulating pitch with an oscillator that runs at audio-rates results in additional harmonics, called 'sidebands', being produced. This leads to a change in tone rather than in pitch.

The nature of these tonal changes depends on the frequency of the modulator oscillator – more accurately, how the frequency of the modulator oscillator relates to the frequency of the modulated oscillator (known as the 'carrier').

Simple, integer-based ratios between the frequencies of the carrier and modulator, such as 2:1, 1:2, 1:3, 1:4, 1:5 and so on, produce 'musical-sounding', or 'harmonic' results. The additional harmonics are musically related to the base oscillator frequency. On the other hand, arbitrary ratios such as 1:2.57, produce more dissonant results.

Cypher2 allows the following FM relationships:

Osc1 **FM 2**: Frequency of Osc1 modulated by the output of Osc2

Osc1 **FM 3**: Frequency of Osc1 modulated by the output of Osc3

Osc2 **FM 3**: Frequency of Osc2 modulated by the output of Osc3

The overall shape of the Osc3 waveform is used as the modulator – therefore there are many parameters that contribute to its output. However, the most important factor is Osc3's frequency, or pitch.

Oscillator FM in Cypher2

Harmonic tuning

Cypher2's pitch control unit/snapping options [LINK - units/snapping] allows each osc's frequency to be specified as either of the following:

• an absolute frequency multiplier of the master pitch, expressed in semitones

• a perfect pitch-ratio of the master pitch, expressed in harmonics (using the Just and Harmonic modes)

The Harmonic mode allows easy definition of harmonic pitch relationships between oscs.

FM depth

Before Osc3 has any effect on Osc2, it is necessary to set the depth of modulation with Osc2's FM 3 control. As the control is increased, more sidebands are introduced – additional tonal partials appear in the frequency spectrum.

1. Set the Osc tuning controls to Harmonic mode by right-clicking on each Osc **Scale** control and selecting *Harmonic* from the context menu.

2. With the master **Pitch** set to 0 semitones, set the Osc3 **Scale** control to *3*. Leave the Osc2 **Scale** control unchanged (*1*).

3. Set the **Out** controls of Osc1 and Osc3 to *0* (all the way down), and fully turn up Osc2's **Out** control.

4. Set the **Wave** parameter of both Osc2 and Osc3 to a triangle shape (move the control fully to the left).

5. Gradually increase the **FM 3** control on Osc2 from *0* to *1* while playing a key.

The pitch ratio of 1:3 (carrier:modulator) is a harmonic ratio – the resulting sounds are musical in nature rather than dissonant. Inharmonic ratios can be useful, however, just like detuned pitching of multiple analogue oscillators. Try setting the Osc3 **Scale** to 3.09. The result still has an overall harmonic tone, but with an interesting dissonance. More extreme inharmonic scaling of the modulator can create bell-type sounds, one of the staples of FM-based synthesizers.

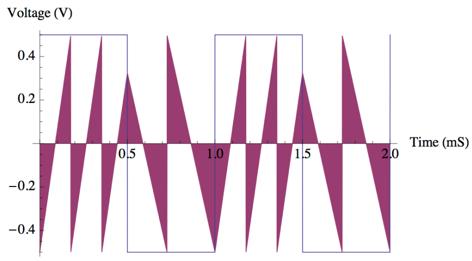
Also note that as the **FM 3** control is swept, the resulting timbral change is not dissimilar to a filter sweep. When using FM, it is possible to achieve a very broad range of tonal characteristics using the oscillators alone. More conventional subtractive synthesizers are heavily reliant on the filter stage for most of their tonal variety.

Thru-zero osc FM

The term 'thru-zero FM' refers to the ability to modulate the frequency of an oscillator beyond zero into negative frequency values. Negative frequencies are vitally important for a stable pitch response with any possible amount of modulation, and is a prominent characteristic of the Yamaha DX-series' implementation of FM / Phase Modulation. Cypher2's implementation provides digital FM-style sounds rendered by a realistic analogue circuit model, with the convenience and polyphony of a software instrument.

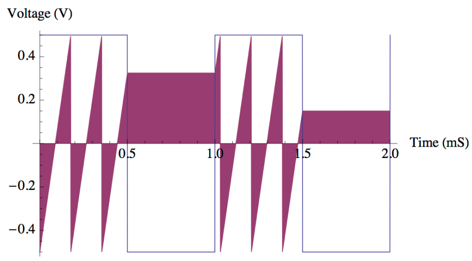
Thru-zero FM is not usually possible with analogue oscillators. When using a typical analogue oscillator, instead of the frequency travelling beyond zero when modulated, it simply stops at zero until the modulation causes it to rise into positive values again. The result is that, at harmonic ratios, the overall pitch that is heard is irregular at lower frequencies. Thru-zero FM allows for the timbre to remain harmonic at all frequencies even with very large modulation.

The following waveform plots show thru-zero and normal analog FM using a square wave as the modulator, and a sawtooth wave as the signal being modulated. While the square wave is high, the pitch of the saw is increased, and when the square wave is low the pitch is decreased.



• Thru-zero FM:

The decrease in pitch results in a negative frequency, and the saw-up waveform becomes a saw- down waveform. A triangle waveform corner is created at the point of transition.



• Typical analogue FM (not thru-zero):

The osc is unable to produce a negative frequency when the pitch decreases. It becomes clamped at zero and stays constant until the modulator increases the pitch. The result is that the resulting pitch is not the same as the original saw wave.

There are currently some thru-zero analogue oscillators available for modular systems, such as the Intellijel Rubicon, Cyndustries Zeroscillator and Doepfer A-110-4 and A-110-6 modules. However, these tend to be rather expensive (especially if polyphony is required!) and there are still inherent problems such as the difficulty of achieving exact tuning ratios and keyboard tracking.

Although thru-zero FM has existed for many years in the Yamaha DX series and its later derivatives, these synths use wavetable-lookup techniques resulting in a very different sound to analogue oscillators. Cypher2's oscillator FM is capable of thru-zero FM using true modelled oscillators generated in real time.

Filter FM

Another classic use of analogue FM is to modulate the cutoff frequency of a filter at audio rates such as that of an oscillator. Cypher2 allows both of its filters to be modulated at audio-rate using Osc3 as the modulator.

Again, this technique is used in order to obtain a change in timbre by modulating the filter cutoff at very fast speeds. Turning up the FM3 control on the filter reduces the action of the filter while introducing additional harmonics, resulting in a buzzy and aggressive sound.

The effect tends to be more pronounced at higher resonance settings, especially as the filter begins to self-oscillate – in such situations, filter FM is effectively like performing FM on a sine wave.

As with oscillator FM, the frequency (pitch) of the modulating oscillator is the main factor in determining the character of the resulting signal.

Thru-zero FM on a filter is inherently impossible in both analogue and digital domains.

Envelopes and LFOs

FM depths become much more interesting when modulated by envelopes or LFOs. The TransMod system must be used in order to achieve this – select an Env or LFO as the source in a TransMod slot and set a modulation depth on the FM controls on Oscs 1 or 2, or in the filter section.

Programming hints

Extreme amounts of audio-rate modulation can be great when wild and experimental sounds are desired but, for most useful, 'musical' sounds, try to use smaller amounts. The same is true of modulating FM depth controls with the TransMod system – for useful rather than crazy sounds, try to find sweet spots within small depth ranges.

Try also looking at the factory Cypher2 presets for ideas and inspiration. Figuring out how sounds have been created using synthesis parameters can be more immediate and practical than studying text-books about mathematical FM theory.

Wave modulation

Wave Modulation uses audio-rate modulation of an oscillator in order to vary its timbre. However, unlike oscillator FM, it modulates the waveform of an oscillator in order to produce these timbral changes, rather than its frequency.

Continuous waveshape of Cypher2's oscs

The waveshape of Cypher2's oscillators can be continuously varied between the following:

• Triangle or Sine (dependent on the state of the **Sine Core** button)

• Saw

• Square

• Pulse

Adjust the Wave control while watching Cypher2's visualizer scope to see exactly how the waveform morphs between the available shapes. The Wave parameter can be modulated by any TransMod source at control rates.

Please note that this cannot be compared to wavetable synthesis - this function provides true morphing between modelled waveforms.

Audio-rate waveform modulation

A specialized function of Osc3 allows its **Wave** parameter to be modulated at audio-rate using the output of Osc2 as the modulator. Increase the **WM 2** control to set the amount of audio-rate waveform modulation. The movement of the oscillator through the available waveshapes at these fast rates causes timbral changes due to the varying harmonics of the waveshapes.

Similarly to FM, the frequency of Osc2 (adjusted using the Osc2 **Scale** control) has a major influence on the sound of Osc3 when using audio-rate wave modulation.

The initial position of the Osc3 **Wave** control also has a large influence on the character of the timbral changes, due to the harmonic differences between the waveforms. For example, a square wave contains only odd harmonics, but as the pulse width is changed (as the waveform changes towards a pulse), more even harmonics are introduced.

Changes in harmonics occur throughout the range of the **Wave** parameter. These changes offer a wide variety of osc sounds, whether modulated at control-rate via the TransMod system, or at audio-rate using Osc3's **WM 2** control.

Wave Modulation and FM

Follow the oscillator FM example in the previous section and then turn up the WM from control on Osc3. Note that the signal becomes more complex and aggressive, with more 'buzzy' harmonics. Try routing an LFO to this parameter so that it changes over time while adjusting the FM from 3 control on Osc2.

When simultaneously using large amounts of the FM and WM controls, the oscs display very chaotic sonic behaviour.

Effects of wave modulation

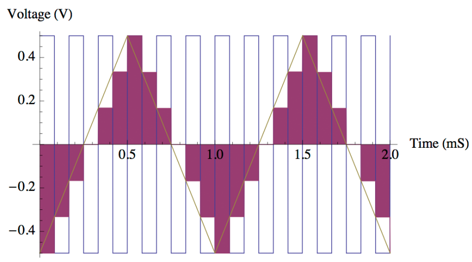
While an osc's frequency (pitch) is the most important factor when performing FM on another osc, its wave shape also has a large influence on the nature and character of the sidebands generated. This is also the case on the modulated osc. Using sine and triangle waveforms typically achieves the most useful 'musical' results. Alternatively, when using the Square/Pulse range of the Wave control (the upper half of its range), the resulting sidebands tend to sound quite aggressive and extreme.

10.2 Sample & Hold, Sync and Beat Detune

Audio-rate sample & hold

Cypher2 features the ability to sample and hold (S+H) an oscillator using the frequency of another, at audio-rate. A sample and hold operation involves the amplitude of one signal being snapshotted ('sampled') on each pulse of a clock signal, and kept constant ('held') at this value until the next clock pulse.

The sample and hold technique is used for a variety of purposes in analogue synthesis, such as generating a random stepped LFO by performing S+H on a noise signal.



In Cypher2, it is used to provide an additional audio-rate modulation technique for achieving increased timbral variation from each osc. The clock pulse is provided by the modulating oscillator's frequency. The result is a quantizing effect on the modulated osc's waveshape.

The sonic characteristics are gritty and buzzy, reminiscent of low-resolution FM and sampling.

The S+H function is accessed by using the **Blend** control on each Osc. This control allows a blend of the S+H signal and the actual oscillator signal.

The S+H position of the **Blend** control should be considered slightly differently to the FM and WM amount controls elsewhere in Osc and Filter sections: the S+H process is always active, with the Blend control allowing the oscillator mix to include the S+H signal.

Here is a summary of what is heard when each osc's **Blend** control is turned fully to the left:

Osc1: Osc2 sampled & held by Osc1's frequency. Osc1 is the clock trigger acting on Osc2

Osc2: Osc3 sampled & held by Osc2's frequency. Osc2 is the clock trigger acting on Osc3

Osc3: Osc1 sampled & held by Osc3's frequency. Osc3 is the clock trigger acting on Osc1

Using audio-rate sample & hold

1. Turn Osc1's **Blend** control to the minimum position. The sound of Osc2, sampled and held by Osc1's frequency, is now the output of the Osc1. At first nothing will be heard if the pitch of both oscs is the same.

2. Turn up the Osc1 **Scale** control while playing a key.

Now the sound of Osc1's frequency applying a sample & hold process to Osc2 can be heard. Lower Osc1 frequencies result in a more quantized Osc2 signal. Many parameters can influence the sound.

Try the following to investigate the effect of these parameters on the timbre (using modulation on the parameters or by manually adjusting controls):

- Change the waveforms of both oscs using the **Wave** control
- Increase the pitch (**Scale**) of Osc2
- Increase Osc3's **Scale** and then increase Osc2's **FM 3** control

Variable-depth oscillator sync

Oscillator synchronization, or osc sync as it is commonly known, is a classic technique found on many analogue synthesizers for producing waveforms with more complex harmonics than those found in basic waveforms such as saw and pulse waves.

Most conventional implementations of osc sync are known as 'hard sync'. This is when the cycle of an oscillator (called the 'sync source' or 'sync master') is used to reset the phase of a second oscillator (the 'sync slave').

In order to create a useful effect, the sync slave osc's frequency should be set higher than that of the sync source osc.

The end result is a more complex waveform from the slave osc that contains additional harmonics, but maintains the same fundamental frequency as that of the master osc.

Main uses of osc sync

A practical advantage of osc sync is that it allows oscillators to be combined without the inherent beating that occurs by simply mixing them (due to the fact that it is virtually impossible to get two analogue oscs perfectly in tune with each other). While the sound of oscs beating against each other is often desirable, there are situations when it can be a problem.

Perhaps the most well-known use of osc sync is to create extreme, aggressive lead sounds. Another common technique is to perform 'sync sweeps' – where the slave osc's frequency is swept manually or with an LFO to create harmonically rich movement.

Variable-depth osc sync in Cypher2

In Cypher2, Osc1 is the sync master, and Osc2 and Osc3 are sync slaves. Both Osc2 and Osc3 can be synced to Osc1's frequency – their phase can be reset on each cycle of Osc1.

However, rather than simply having 'sync on/off' buttons for the slave oscs, Cypher2 features variable controls, called Sync to 1, for these two oscs.

• Minimum (0) and maximum (1) settings

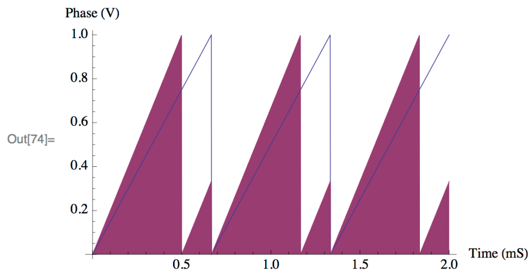
When **Sync 1** is turned up to its maximum setting (100%) on Osc2 or Osc3, the phase of these oscs is reset fully to their initial state upon each cycle of Osc1. This technique is commonly known as 'hard sync'. When the control is at the minimum setting (0), no osc sync occurs.

• Settings between 0 and 1

At **Sync 1** settings between 0 and 1, Osc2 or Osc3 reset their phase on each Osc1 cycle only if the following condition is met:

Osc2 [or Osc3] phase > (1- (Osc2 [or Osc3] **Sync 1** amount))
/100

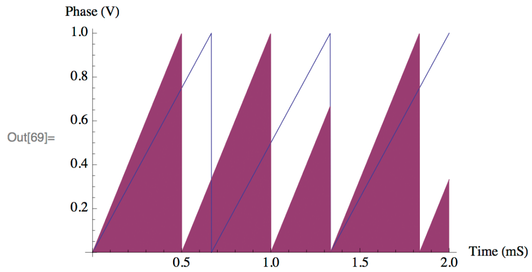
This resembles some analogue implementations of 'soft sync', and can create a range of complex sounds that are not possible with full, 'hard' osc-sync.



• **Sync 1** control at 100%

This example shows a saw-up waveform on Osc2, synced to a lower-frequency saw-up waveform on Osc1. The Sync to 1 control is set to 100%.

Note the phase of the Osc2 waveform is reset to 0 on every completed cycle of Osc1.



• **Sync 1** control at 50%

In this example, the Osc2 waveform does not re-sync on the first cycle of Osc1, because Osc2's phase is approx. 0.35, which is not greater than 0.5 (or 1-(50/100)).

On the second cycle of Osc1, Osc2's phase is approx. 0.67, which is greater than 0.5 – therefore Osc2's phase is reset to 0 and it syncs to Osc1.

Using variable-depth osc sync in Cypher2

1. Change each osc's unit/snapping options to *Equal*.

2. Fully turn up Osc2's **Gain** control.

3. Play a key – a simple saw-wave tone is heard.

Now increase the Osc2 **Scale** control to *+15* and play a key. This time, a 2-note minor chord is heard instead..

4. Increase Osc2's **Sync 1** control.

As the setting goes over approx. *30%*, the chord is no longer heard – it is now a complex waveform.

5. Now try a 'sync sweep': move the Osc2 **Scale** control up and down.

6. Try the above at different settings of the **Sync 1** control to explore the timbres that are possible.

Beat detune

When two detuned oscillators play together, their differing frequencies cause an audible 'beating' effect against one another. The rhythmic rate of this beating varies depending on the notes played, since beating is directly affected by the frequencies of the notes themselves - different notes generate different frequencies. While it is common to use such sounds in a rhythmic way, any performance is always influenced by different note pitches each resulting in a different beating rate.

Each of Cypher2's oscillators includes a useful feature for providing this detuned beating effect with the same rate across the whole keyboard range, resulting in new ways of performing throbbing basses and rhythmic pads / chords.

Using each osc's Fine parameter to create beat-detuning effects

Each osc's individual **Fine** control sets the frequency rate of the beating effect in Hz. To understand how this control works, follow this simple example:

• Typical detuning

Start with an initialized Cypher2 state. Osc1 and Osc2 are both set to the same frequency in relation to the master Tuning section's **Pitch** and **Fine** parameters. When played together normally, with each osc's **Fine** controls set to zero, there is no beating – the oscs are perfectly in tune with each other.

Now, set the Osc2 **Scale** control to 1.04 (in other words, +4 cents or +0.04 semitones). This creates beating at a rate of approximately 1 Hz when playing the A4 key on the keyboard.

This is because this pitch, when combined with keyboard tracking, causes the A4 key to produce a frequency of approximately 441 Hz from Osc2. When this osc frequency plays at the same time as Osc1 at exactly 440 Hz, the difference between each osc's frequencies is the rate of beating.

However, if the A5 key is played, the beating rate would be approximately 2 Hz. This is because, with keyboard scaling, each increasing octave doubles the oscillator frequency: the Osc2 frequency is 882 Hz, while Osc1 is at 880 Hz. The difference between them is 2 Hz, which is the beating rate.

• Beat detuning

Now set the Osc2 **Scale** control back to 1 and set the Osc2 **Fine** control to 1 Hz. This results in a beating rate of 1 Hz across the whole keyboard, because the 1 Hz is added after the keyboard scaling is applied:

Osc2 at A4: 440 Hz + 1 Hz = 441 Hz Osc2 at A5: 880 Hz + 1 Hz = 881 Hz

Because the difference between the frequencies of the two oscs is the same, the beat rate is also the same.

With the **Beat Detune Sync** button activated, the **Fine** control is set in BPM-based values synchronized to the host tempo, allowing tempo-synced beating effects.

A dedicated beat rate/detune control is very rarely found on hardware analogue synths, with a notable exception being the Moog Taurus bass synthesizer.

Beat rates as TransMod modulators

The beating rates of the various oscillator combinations are available as TransMod control-rate sources. This means that other parameters can be modulated at beating rates, whether using the Fine controls for constant-beat detuning or not.