KARMA·FX

SYNTH MODULAR 2

User's Manual

Revision 0.36

KarmaFX Synth Modular Version 2.01+
VSTi / VST for PC / Windows™ and Audio Unit / VSTi / VST for Mac / OS X.
Created by Kasper Nielsen,1998-2021.

E-mail: karmafx@karmafx.net
Web: https://karmafx.net

Software License Agreement

1. Grant of License and Copyright: KarmaFX grants you, the original purchaser, the non-exclusive and non-transferable license to use the program and data files constituting this software and the upgrade programs(s) and the data file(s) which may be distributed to you from time to time (herinafter called the "Software") on up to two computers which are under your control. You are allowed to use the Software royalty-free for music and sound production, recordings and/or performances. The ownership of the rights to and the copyright of the Software (whether supplied via a CD, diskette, download via the internet or otherwise) belong to KarmaFX. KarmaFX reserves the right to make modifications to the Software and to any support services without prior notification.

2. Restrictions: The Software contains copyrighted information. For the purpose of protecting such copyrights, you may not sell, sublicense, rent, lease, convey, distribute, copy, modify, translate, reverse engineer, convert to another programming language, decompile, or disassemble the Software for any purpose, or make derivative works based on what is contained in the Software. The Software is Registered in your name and may only be used by you personally. You may make a reasonable number of backup copies of the Software for your own personal use only. You may not transfer the Registered Software to another computer through a network or to another party by any means. As a condition to use the Software, you agree that you must not use the Software to infringe the intellectual property or other rights of others, in any way. The unauthorized reproduction, distribution, modification, public display, communication to the public or public performance of copyrighted works is an infringement of copyright.

3. Termination: This agreement takes effect on the date you have received the Software. The license granted in this agreement will terminate automatically without notice from KarmaFX in the event that you breach any provision of this agreement or in any way infringe KarmaFX's copyright in the Software. In such event you must destroy the Software and all backup copies immediately.

4. Disclaimer: The Software is provided as-is, and KarmaFX disclaims any warranty or liability obligations to you of any kind. KarmaFX disclaims all warranties relating to the Software, whether express or implied, including but not limited to any implied warranties of merchantability and fitness for a particular purpose, and all such warranties are expressly and specifically disclaimed. KarmaFX shall not be liable for any indirect, consequential or incidental damages arising out of the use or inability to use the Software even if KarmaFX has been advised of the possibility of such damages or claims. The user of the Software bears all risk as to the quality and performance of the Software. KarmaFX will not be liable in any way for any allegations or findings of infringement of copyright or other proprietary rights as a result of your use of the Software.

Contents

1. Introduction

Thank you for purchasing KarmaFX Synth Modular!

KarmaFX Synth is an advanced simulated analog modular synthesizer that can be used in the studio either as an instrument or as an effect. It delivers superb sound quality and offers powerful modular flexibility while still being very easy to use.

Like a true modular, the Synth consists of modules that can process audio and control signals and be patched into other modules. Needless to say, this offers far more versatility than a fixed path synthesizer that only allows you to adjust predefined knobs.

Hi and Welcome ! Throughout the manual small text boxes like this one will explain words underlined in the text that may be hard to grasp.

This manual is intended to give you a basic understanding of how the Synth works. It describes the functionality of every available module in detail. After reading this, you should be able to roll your own patches with ease. A basic understanding of fundamental sound synthesis is recommended but not required.

The KarmaFX Synth is a VST plugin, meaning that it will run inside any VST-compatible host application on PC/Windows. It has been tested and verified to run in many different hosts, such as Steinberg Cubase, Ableton Live and Image-Line FL-Studio, just to name a few. The synth is also available as Audio Unit and VST on Mac/OS X. Audio Unit compatible hosts for Mac include software such as Apple Logic and Ableton Live..

VST is an acronym for Virtual Studio Technology: A plugin interface developed by Steinberg Corp.

Audio Units

In section 2 you will find a brief overview of the Synth's features and its system requirements. It also contains a detailed run-down over the most important new feature introduced in version 2. Section 3 shows you how to install KarmaFX Synth in your favorite host. Section 4 introduces the general concepts of modular synthesis. Section 5 and 6 describes the user interface and how to operate the Synth. Finally, section 7 is a reference that lets you dive into the details of every module.

So go right ahead… read on to learn the secrets of how to tame this modular beast. Then go write a killer track! ☺

We at KarmaFX sincerely hope that you will be happy with your purchase - and fall in love with synthesis all over again

2. Features & Requirements

2.1 Feature Overview

Modular synthesizers are remarkable pieces of machinery. For readers who cannot wait to get down to the gritty details, here is a list of feature highlights in KarmaFX Synth Modular:

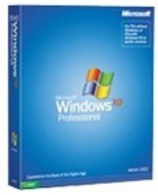
- Advanced Simulated Analog Modular Synthesis: Modular patching of synth modules and internal high-frequency digital simulation of analog voltage levels.
- Oscillator with Phase, Detune and Pulse-width that simulates standard analog waveforms. Dual Oscillators with Hard-Sync and Ring Mod.
- 16/24/32 bit mono/stereo Multi Sampler that imports `WAV`, `SF2` and `SFZ` files. Key/Velocity-ranges and Loop-Point Multisample editing.
- Additive and Pad Module with waveform and harmonic magnitude & phase editor. Up to 1024 harmonics + support for user-defined presets.
- Granular Module that offers sample-based granular synthesis, with variable grain size and rate and modulation/diffusion options.
- 2/4 Pole Multimode Filters with Cutoff, Resonance (LP, HP, BP and BS) + Saturation and Drive: SVF, SVF2, SKF, Zolzer, Moog, Moog2, Acid, MS20. 3/10/31 band Equalizers, Formant, Comb, Allpass, Parametric and Shelving.
- Amplifier and Stereo modules with panning, volume and velocity controls. Two channel Mixer with Ring modulation and bit operations. Inverter for Amplitude Modulation, and Mid/Side for stereo separation.
- Delay, Reverb, Phaser, Chorus/Flange, Pitchshift, BitShuffler and Distortion effects, Folder, PanSpread, Soft Clipper, Maximizer, Soft-knee Compressor and Multiband Compressor with Peak/RMS detection.
- 10-Octave / 12-Note Pitch control with detune and portamento. Choose between mono, legato, or up to 16-voice true polyphony Controllers for frequency and phase modulation (FM/PM). Up to 16 channel Unison controller with detune and stereo pan spread.
- Support for Linear FM, Phase Modulation, and Exponential FM with Exponential Frequency Sync. Generators support Through-Zero frequency, and optional High Quality (HQ) 16x oversampling.
- Scope Module with multi-layered, synced waveform oscilloscope, and frequency and phase displays for detailed sound inspection.
- Pattern controller with 1 to 32 steps, 1 to 4 octaves, hold, loop, and legato support and 8 user programmable pattern presets. Can work in Step and Arpeggiator mode, with optional scale-based note-masking.
- Bipolar/Unipolar LFO, HFO, ADSR and Multipoint Envelopes, Step Sequencer, Env.Follower, S&H, Decay, Shaper and Control modules.
- Expression, so parameters can respond to MIDI Mod.Wheel, Aftertouch, Velocity and Timbre inputs + MIDI Polyphonic Expression (MPE) support.
- Control Panel & Patch Browser for quickly navigating patches and banks.
- Keyboard module for MIDI visualization and manual MIDI triggering.
- Output module with Panning, DC removal, volume and clip control.
- Input module so that the Synth can function as an insert effect.
- Noise generator, filtered pink, white and brown noise with freq. sync.
- Full stereo support (modules can run in mono to save CPU cycles).
- Instant visual feedback of waveforms, modulation and controls.
- Up to 128 simultaneously running internal voices (Polyphonic/Unison).
- 128 user-assignable automation controls with MIDI Learn.
- SubPatch generator and effect modules for re-using patches within other patches, and doing complex generator/effect patch construction.
- 5 banks of pre-made KarmaFX patches + Extra user-banks. More than 1000+ patches total. Additional Online Banks can be downloaded and installed from inside the synth.
- Synth frequency can be offset from "Concert Pitch". (e.g., 440Hz → 432Hz)
- Fast output response, with zero latency and tight latency compensation.
- Fully skinnable GUI: Skins bundled with the installation support HiDPI/Retina GUI scaling of up to 200%.

2.2 System Requirements

In order for KarmaFX Synth Modular to run properly, the computer system must fulfill the following requirements:

PC / Windows:

- A 2 GHz Pentium class CPU with SSE (Intel or AMD) is highly recommended. Absolute minimum is a Pentium class CPU running at 1 GHz, 1 GB RAM (4 GB RAM recommended), 200 MB free disk space, Minimum 1920 x 1080 screen resolution, 32 bit color.
- Operating system: Windows XP/7/8/10.
- VST 2.4+ compatible host application.
- Low latency sound card (preferably with ASIO driver).



Mac / OS X:

- Mac computer with an Intel processor and 200 MB free disk space.
- Operating system: OS X 10.9 or later.
- Audio Unit (v2) or VST 2.4+ compatible host application.

2.3 What's new in Version 2

For readers already familiar with KarmaFX Synth Modular, this sections offers a run-down over some of the most important additions and changes that version 2 offers.

- Version 2 is backwards compatible with version 1.x, meaning that all existing patches and patch-banks will load in version 2. In most cases they will also sound the same, but due to the many internal changes this is not 100% guaranteed.
- A set of new UI features have been added, including circular-knob-modulation-meters that show the current knob modulation, 20% larger UI overall, drop-shadows, glowing displays, gradient curves, semi-transparent menus and motion-blurred trail markers. Three brand new skins have been added, that support high quality scaling for HiDPI and Retina displays, while the classic Blue skin is also still available.
- A new **Granular** module (p.57) has been added that offers sample-based granular synthesis, with variable grain size and rate and modulation/diffusion options.
- A new **Patch Browser** (p.35) has been added, which allows for quick browsing of banks as well as patches, and also offers keyboard browsing and keyboard filtering.
- Version 2 introduces *Subpatches*: The **SubPatch** generator and effect modules (p.102) allows for patches to exist and work within other patches. This is a powerful feature which means that existing patches can be reused inside other patches, and that modules can be grouped into subpatches for more elegant wiring. Two levels of subpatches are supported, which for all practical purposes means that the synth per patch now offers an unlimited amount of modules.
- New Effects modules have been added: A **Folder** module (p.91) that performs wavefolding through classic 0Hz Frequency Modulation, a **SoftClip** module (p.100) that offers variable softclipping, a **Repeater** module (p.101): a beat-synced, repeated delay, a **Limiter** module (p.99) that offers soft-knee peak limiting, and the **MultiComp** module (p.93)**:** a 3 band multiband soft-knee compressor.
- New Controller modules have been added: The **Phase** module (p.76) which offers control over the phase control signal, and complements the updated **Frequency** module (p.75), a **Scope** module (p.83): a synced oscilloscope, frequency and phase analyzer of up to two separate input signals, as well as all internal controls signals, and a MIDI **Keyboard** trigger/visualization module (p.84). **Unison** has been reworked to offer more modes as well as **Phase Detune** (p.79) and Pattern module

has been updated with **Steptime**, **Velocity** and **Min**/**Max Note** range-settings (p.80). **Notepitch** & **Pattern** modules can now be detuned from *Concert Pitch* (A=440Hz).

- New Amplifier modules have been added: An **Inverter** module (p.72) for audio and modulation inversion and a **Mid/Side** module (p.73) for mono/stereo separation. All panning operations are now constant powered, -3dB.

- Filters and been reworked in part to be virtual analog, which gives better stability and allows for higher frequency filter modulation. Three new filters have been added: The **SVF2**, the **SKF** (Sallen-Key Filter) and the **Moog2** (p.60), plus all Resonant Multimode filters now have **Saturation** and **Drive** controls.

- An **Exponential Tuning** of all Filter-Cutoffs has been added (p.60), giving better control over both high and low- filter frequencies, which sounds more natural when modulated. A **Parabolic Tuning** option is available for backwards compatability. Similarly, a **Resonance** tuning option has been added (p.60), featuring a flat **Digital** response and a simulated **Analog** response that attenuates resonance at low frequencies. The Filters **Keyboard Tracking Base Key** is now also adjustable.

- New Modulator modules have been added: A **Shaper** module featuring customizable waveshaping of modulation signals (p.115), and a **Control** module that exposes the internal control signals for use as custom modulation sources (p.116). **LFO** now supports optional **Random Poly Phase** Reset and **Keyboard Tracking**.

- All Generators now support so called **Through-Zero** frequency, meaning that they accept negative frequencies that generate output in reverse phase. An extra **Phase Control Signal** has been added, to complement the existing Frequency, Trigger, and Note Control Signals. This means better control over phase, which is useful for phase modulation. The FM module has been reworked to support **Linear FM** and **Exponential FM**, with **Through-Zero** support as well as **Phase Modulation** (p.77).

- All Control Signals can now be 16x oversampled in **High Quality** (HQ) mode, forcing generators to run oversampled internally. Essential for high quality FM/PM.

- An **Expression** option has been added to knobs that typically need performance tuning, such as Cutoff, Resonance, Modulation Index, LFO rate & amount, Amplification, etc. responds to Mod.Wheel, Aftertouch, Timbre and Velocity inputs.

- To quickly clean-up patches, the right click menu now offers an **Auto Arrange** option that neatly re-orders modules, and a **Remove Unused** option that deletes disabled or disconnected modules from the patch. A **Swap** menu option now also quickly swaps modules backwards and forwards in the signal chain. Finally, **Insert** offers a quick menu-based way to wire a specific module in front of another module.

- Control Panel automation controls Min- and Max-values can now be *macro* programmed, to limit parameter values to user-defined ranges (p.34).

- Many parameters now have extra **Range Menu Options** added to their right click menu, allowing for more fine-grained control over, e.g., Attack, Decay, and Release time-ranges in all envelopes, all Portamento time ranges, Modulation Index ranges, Delay Finetune ranges, LFO rates, Chorus Depth ranges, etc.

- Modules can now be connected using a simple one-click mouse-action + connections now highlight when mouse-hovering in connect mode.

- A set of new patch banks have been added. Patch banks can now be read-only, and loaded while compressed on disk. New patch banks can now also be downloaded and installed instantly from within the synth using the **Bank Install** menu, available under Options (p.31).

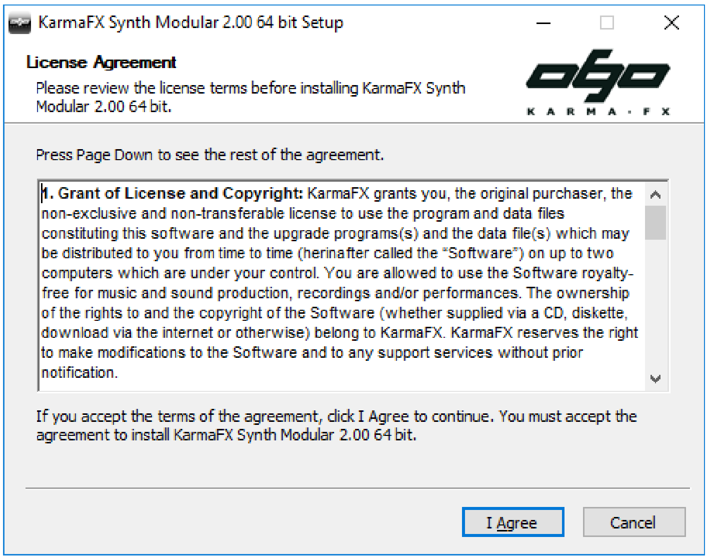
These additions are just the most prominent features. Many more minor improvements, and optimizations have been added in version 2. Please refer to the full version 2 changelog online for more technical details.

3. Installation

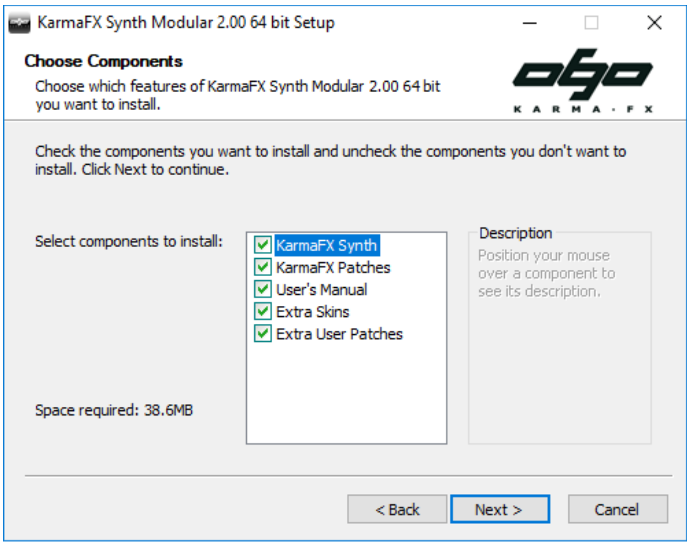
3.1 Installation on PC/Windows

This section will show you how to install the Synth on your Windows PC. Make sure to close all running sound applications before starting the installation. Then…

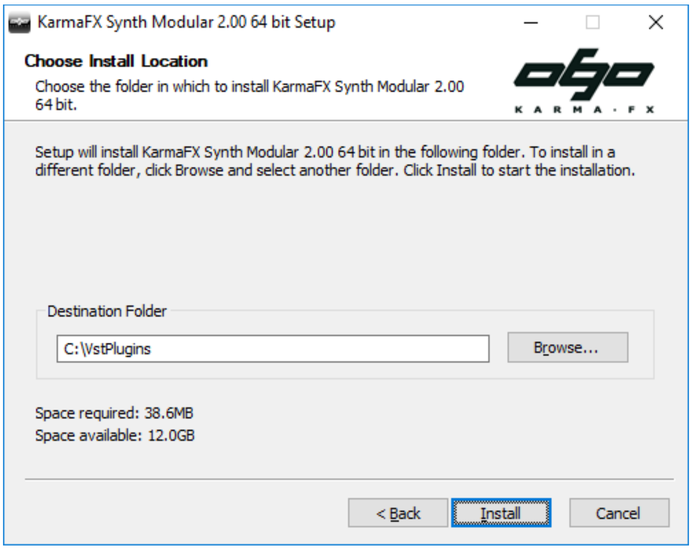
▶ Locate and run the the installation file. Use **KarmaFX Synth Modular 2 32 bit** or **KarmaFX Synth Modular 2 64 bit** for either 32-bit or 64-bit installations respectively.



▶ After reading the License Agreement, click **I Agree**. This brings up the installation selection menu:



▶ Everything to be installed is checked by default, so simply click **Next** to proceed.



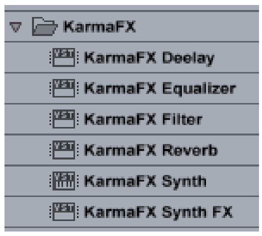
► Now select the VST-folder where your host application's VST-plugins are placed. In order to save patches inside the synth, it is important that this folder is user-writeable. We suggest:

`C:\VstPlugins`

Some hosts may require a different path, but in most cases you can choose it yourself. Please refer to your host application to get the required folder path, if any.

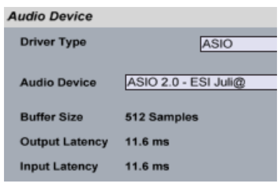
Click **Install** and the Synth will install into that folder. Afterwards, click **Close** to end the installation program.

►Start you host application. The host will usually scan for new plugins on start-up. If this does not happen, make sure to re-scan for VST-plugins. A **"KarmaFX Synth"** and **"KarmaFX Synth FX"** plugin should appear under VST-instrument and VST-effect, respectively:



►Finally, make sure your host applications sound output latency is set as low as possible. For best results, use an *ASIO driver* if possible and set the latency buffer to 128, 256 or 512 samples. The latency buffer is filled with sound samples before they are output to the sound card. This means: The longer the buffer, the longer the latency, or delay, before the sound reaches your ears. Small latency is better, since this means that the time it takes in milliseconds from a key hit on your keyboard and until your actually hear the sound, is insignificant.

ASIO (Audio Stream Input Output) is a protocol for low-latency digital audio specified by Steinberg.



That's it! You are all set.

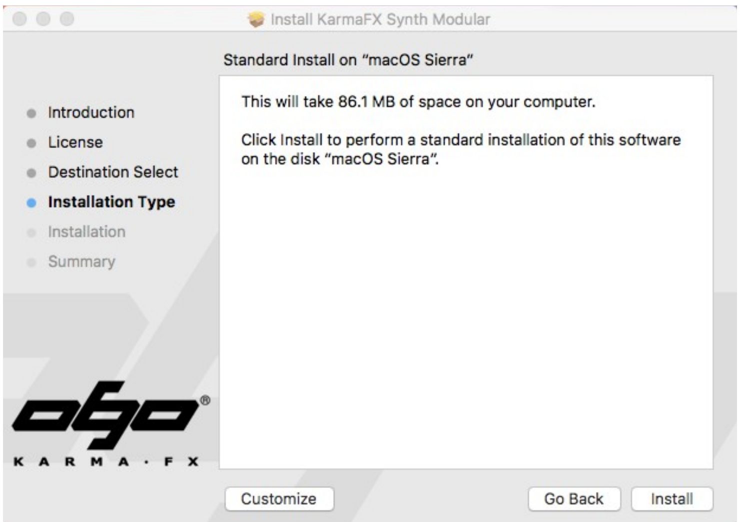
This manual in PDF format is located in:
`<vstfolder>\KarmaFX\KarmaFX_Synth\Manual`

3.2 Installation on Mac/OS X

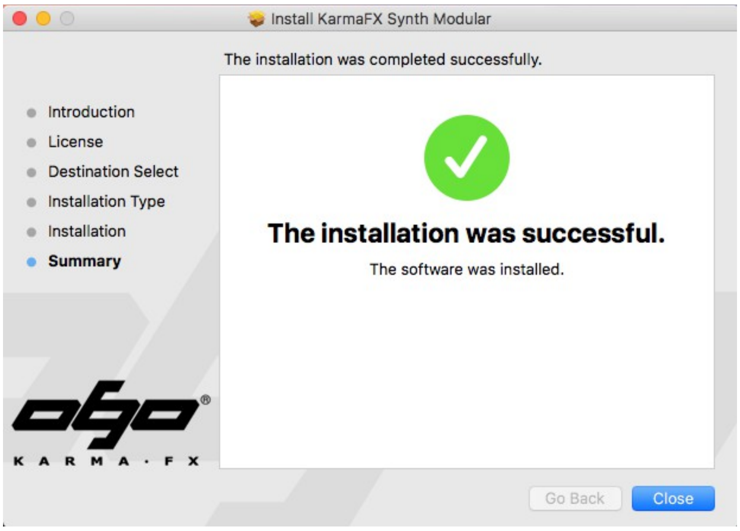
For installation on Mac/OS X, first make sure to close all running sound applications. Then...

▶ Locate and run the supplied PKG installer and follow the instructions. Use **KarmaFX Synth Modular 2 32 bit** or **KarmaFX Synth Modular 2 64 bit** for either 32-bit or 64-bit installations respectively.

▶ Click **Continue** to agree to the License Agreement. After that the installation files can be customized by clicking the **Customize** button. However, everything to be installed is checked by default, so simply click **Install** to proceed with standard installation.



▶ Once the installation completes, you will be presented with the "The installation was successful." message shown below. This concludes the installation and you can simply click **Close**.



▶Finally, you may start your sound host application. And that's it! You are all set.

The host will usually scan for new plugins on start-up. If this does not happen, or the synth doesn't show up, make sure that it is a VST- or Audio Unit compatible host and that Audio Units support is enabled. Rebooting your Mac may in some cases be needed to force a rescan of the Audio Units folder.

The KarmaFX Synth Modular Synth & FX Audio Units are called:
`KarmaFX Synth.component`
`KarmaFX Synth FX.component.`

After installation, they are located in:
`/Library/Audio/Plug-Ins/Components/`

VST plugins are located in:
`/Library/Audio/Plug-Ins/VST/`

The core files for the plugins, patches, presets etc. are stored in:
`/Library/Application Support/KarmaFX`

This manual in PDF format is located in:
`/Library/Application Support/KarmaFX/Manual`

4. Concepts

This section will give some general background information on modular synthesis and explain the basic concepts used by the Synth. This is important in order to understand how the Synth produces and processes sound internally.

4.1 Digital Simulation of Modular Synthesis

Back in the days when analog synthesis thrived, a hardware modular synthesizer was considered state of the art for sound synthesis – and it still is in many ways.

A modular synthesizer is a system consisting of small devices, called modules, each solving their own specific task. The modules are wired together by cables. The cables carry a signal: an electric current of variable voltage, from a module's output to another's input. The signal can be an audio signal or a control signal.

Control signals are non-audio signals used to control certain variable parameters.

Some standard modules found in modular synths are:

VCO: Voltage Controlled Oscillator, which produces a waveform at a specified pitch.

VCF: Voltage Controlled Filter that cuts away frequencies, changing a sound's timbre. Hence, this is usually called *subtractive synthesis*.

VCA: Voltage Controlled Amplifier, which controls the amplitude or overall volume.

ADSR: Envelope generator (abbreviation for Attack, Decay, Sustain, Release) that simulates the contour of a natural decaying sound like a piano. Can be used to modulate, e.g., the VCA or the VCF to control the amplitude or the timbre of the sound over time.

LFO: Low Frequency Oscillator, which outputs a low frequency waveform, usually a sine or a triangle wave. Normally used as a control signal, e.g., for changing the pitch of the VCO to create a vibrato effect.

VCO/VCF/VCA section in Roland's System 110m

Many different modules exist, but the real strength is that modules can be wired together in completely arbitrary ways, making the sound creation possibilities almost endless.

Typically, a modular synthesizer is controlled by a keyboard. So playing a melody on the keyboard's keys will play the created sound accordingly at varying pitch. This is done by controlling the modules frequency (e.g., VCO) by a connected keyboard using two signals: A *Frequency* or *Keyboard Control Signal* that controls the module's frequency depending on the pressed keys, and a *Trigger Control Signal* that tells when a key is pressed and released.


Moog Modular

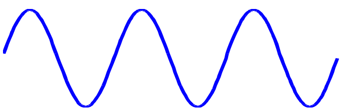
Historically, the first modulars had real physical modules that the musician connected with real cables to construct a sound. One of the first modulars was the *Moog Modular Synthesizer*, designed by Bob Moog back in 1963. Later came modular synthesizers from other manufacturers (*ARP, Serge, Buchla, EMS* and the *Roland System 700*). Although they were expensive at first, musicians and sound designers were awe-struck by the ability of these machines to create new unique sounds as well as imitate existing instruments.

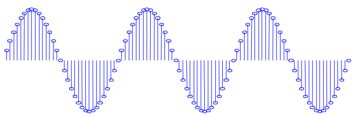

ARP 2600

Modular synthesizers have since had a great impact on music evolution. Especially electronic music has been influenced by their capability to create sounds that do not necessarily sound like ordinary instruments.

KarmaFX Synth simulates the exact same signals as in an analog modular synth, but in a digital system, namely the computer. The difference between an analog and a digital signal is simply that the digital signal is a sampled version of its analog brother, i.e., it is made up of a series of numbers instead of a continuous voltage.

Sample rate is the number of samples per second


continuous analog signal


sampled digital signal

The sample rate determines the quality of a digital signal, i.e., the higher the sample rate the better the approximation to the analog world. For sound, a frequency of 44100 samples per second (44.1kHz) is normally considered adequate for human hearing. For the best possible quality, KarmaFX Synth internally uses the same high frequency for both sound and control signals.

In fact, the Synth does not distinguish between audio and control signals. They are treated the same and can be used interchangeably like in a true analog modular system.

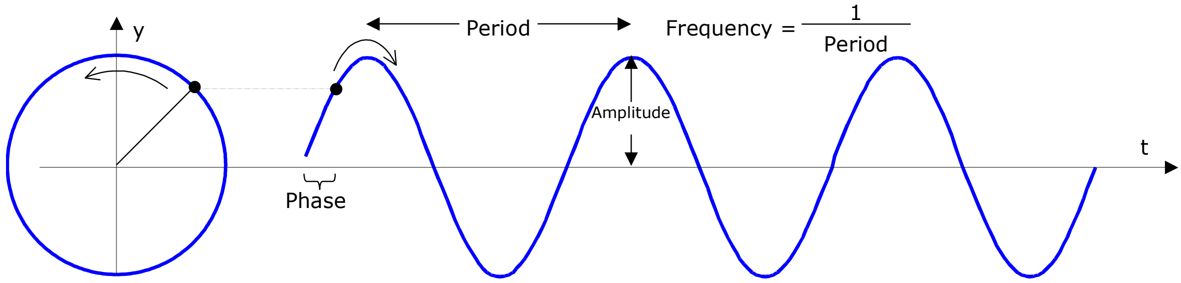
4.2 The Art of Sound Synthesis

This section gives a brief introduction to the fundamental principles of sound synthesis. If you are already familiar with *sine waves, additive* and *subtractive* synthesis, and words like *amplitude*, *frequency* and *phase*, you may skip this section entirely.

The word sine is derived from the Latin word sinus, which means "bay" or "fold".

Sine waves

Sine waves are a fundamental component in synthesis. In fact all sounds can be built up of sine waves at different speeds. The construction of a sine wave can be perceived as a point swirling around on a circle and plotting its *y* value over time:



The time it takes for the point to do a full cycle is called the *Period*. It tells us the speed of the sine wave. Another measure for this is *Frequency,* which is simply *1/Period*. Frequency is in fact equal to the number of cycles the sinusoid does per second.

The size of the sine wave is called the *Amplitude.* It measures its general loudness or air pressure.

Last but not least, where we start our cycle on the circle can be of importance. This is called the *Phase.*

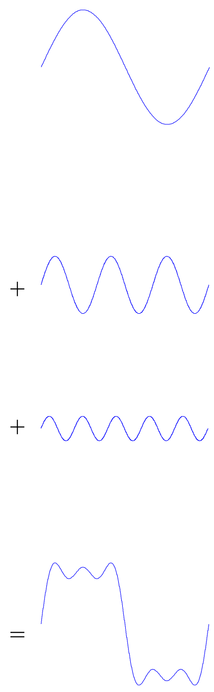
So, a sine wave can be described using the three parameters: *Frequency, Amplitude* and *Phase*. In mathematical terms:

*Sine(t) = Amplitude * sin(2*PI*Frequency*t + Phase)*

PI is the name of the Greek letter π, pronounced "pie". It is a constant approximately equal to 3.14159.

If we were to exchange the *sin* function in the expression above with some other function, this formula can in fact be used to describe any periodic (repeating) signal.

Additive Synthesis

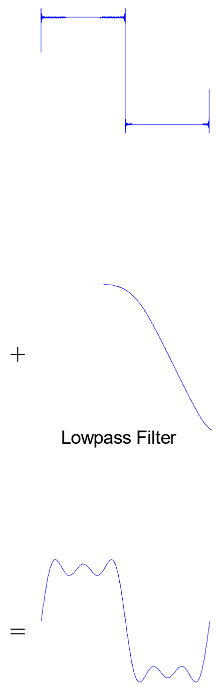


Additive synthesis is a technique that generates sounds by summing several sine waves at different amplitude, frequency, and phase. The idea is that since all sounds can be decomposed into sinusoids, all sounds can (in theory) be synthesized using sinusoids too.

Any sound produced by an instrument typically consists of a periodic waveform starting with a fundamental sine wave. This sinusoid has the lowest frequency of the sound and dictates the resulting sound's musical pitch. Any additional sine waves have a frequency that is an integer multiple of the fundamental frequency. These sinusoids are called the sound's *harmonics*.

An additive sound's harmonics phase, frequency and amplitude can be changed over time to make the sound more interesting. Drifting sounds are often perceived by humans as beautiful, and additive synthesis is therefore good for modeling pads, evolving textures and ambiences.

Subtractive Synthesis



Subtractive synthesis is the exact opposite of additive synthesis. Here a sound is created by starting with a complex waveform and then modifying this harmonically rich sound by using filters to attenuate and/or boost frequency content.

The common Lowpass filter has a cutoff frequency and resonance setting. The cutoff frequency determines at which frequency the filter should start to attenuate, while the resonance controls the amount of boost to apply near the cutoff frequency. This is good for simulating the natural timbre of many instruments.

Typically, the filter is varied to change the timbre of sound over time. Likewise the amplitude can be changed to simulate the rise and decay of the sound.

Historically, subtractive synthesis gained popularity due to its use in analog voltage controlled synthesizers. A simple circuitry can be used to generate the common sawtooth, square and triangle waveforms, all rich in harmonic content. It is simple to control and can produce a wide variety of sounds.

Sample Synthesis

Sample synthesis is a technique that uses recorded sounds stored as digital samples. The idea is that instead of trying to re-create the sound of a real instrument, why not just record the real thing and play it back?

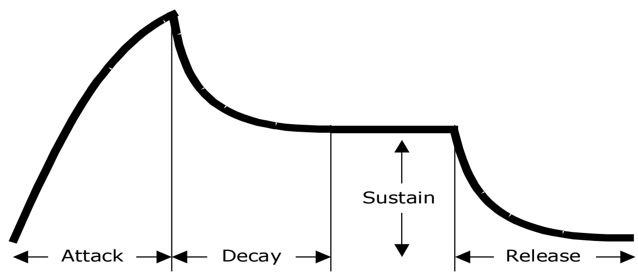
Typically, several sampled sounds have been recorded at pre-selected frequencies. They are then played back at different frequencies using interpolated resampling. Finally, the samples are altered by filter and amplitude changes in order to create a realistic imitation of an instrument. Samples are either single cycle waveforms (Wavetable synthesis) or larger recordings of instruments. The latter is a natural choice, e.g., for percussion.

Granular synthesis is a variation of sample-based synthesis, where the sample is chopped up into tiny sample clips, called grains. These are then re-assembled and played back overlapped using a windowing function. Unlike regular sample playback, granular synthesis allows the frequency to be controlled independently from the sample playback speed.

Synthesis by Modulation

If we look at the synthesis techniques above, it is clear that they all share a common concept, namely *Modulation,* or *changes* in sound. Modulating sound parameters using control signals is important to make sounds come to life, and a key feature in modular synthesis.

Typically, a parameter is modulated by an *envelope* signal. When a key is hit, e.g., on a piano, the sound amplitude rises quickly to a certain level, then fades a bit, until the key is released and the sound dies out. This shape of the sound is called an Envelope. The most common is the ADSR, which is an abbreviation for *Attack, Decay, Sustain, and Release*.



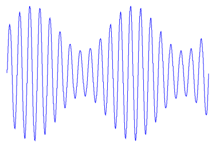
As shown, Attack, Decay, and Release, control the time for rising and falling of the envelope level, while Sustain controls the level when a key is held.

So, if we apply this envelope to a sound, we are in fact modulating the sound’s amplitude. This concept can be extended further to modulate all kinds of parameters. Here is a list of common modulation types:

Amplitude Modulation (AM): Envelope amplitude modulation is used to shape a sound. Low frequency periodic Amplitude Modulation creates a *tremolo* effect.

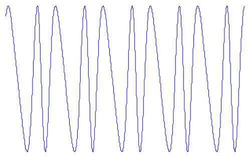
Tremolo is a modulation effect created by varying the amplitude, or volume, of the signal.

Frequency Modulation (FM): The source (*Carrier*) signal's frequency is modulated by a, normally periodic, modulation signal (*Modulator*). This changes the timbre of the sound resulting in a more complex waveform by creating so called *sidebands*. When creating harmonic sounds, the frequency of the Modulator must have a harmonic relationship to the Carrier, i.e., if the frequency of the carrier is n, then the Modulator's frequency could be 2n, 3n, 4n og ½n, etc. However, frequencies that are non-integer multiples of each other (non-harmonic) can also be used, e.g., for creating bells and percussive sounds. Low frequency FM gives the musical impression of *vibrato*.



Amplitude modulated sine

Phase Modulation (PM): At low frequencies, Phase Modulation works like Amplitude Modulation. At high frequencies it gives similar results to Linear Frequency Modulation.



Frequency modulated sine

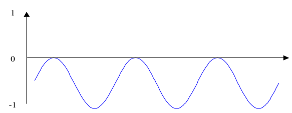
Pulsewidth Modulation (PWM): For periodic waveforms like square and triangle it is possible to control when the waveform changes sign; the waveform's so called *duty-cycle or pulsewidth*. Modulating the pulsewidth, i.e, changing the fraction of time the signal is "on" instead of "off", gives a result similar to adding many identical waveforms running slightly out of phase, resulting in a fatter sound.

Filter modulation: Modulating, e.g., Lowpass, Highpass, Bandpass filter parameters (like the Cutoff frequency) is often used to open or close a sound over time. A filter can also be forced to (more or less) follow the frequency of the frequency control signal (*keyboard tracking)* in order to keep the same timbre regardless of the waveforms pitch.

Ring modulation: Ring Modulation multiplies a source signal with a modulation signal (typically a sine wave). The resulting output is the sum and difference between the two signals; good for creating bell-like and metallic sounds.

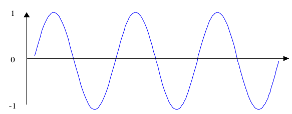
Ring modulation got its name due to ring-shaped diodes used in the analog circuit that originally created the effect.

We normally distinguish between two kinds of control signals: *Unipolar* and *Bipolar*. Unipolar has always all positive or all negative amplitude values.



Unipolar signal

Bipolar has both positive and negative values centered around 0. Typically, modulation signals are unipolar, while sound signals are bipolar.



Bipolar signal

Sound Effects

So far we have looked at additive/subtractive/sample synthesis as a means to generate sounds and modulation synthesis as a means to alter sound characteristics. But sound effects such as delay, reverb, distortion, chorus and pitch-shifting can quite heavily alter sound characteristics too, and are, therefore, commonly used tools in modular synthesis.

Delay (echo) simply copies an incoming signal and outputs it at a later time. While simple in concept it can add depth and space to a sound, especially if the effect is combined with stereo and filter changes.

Reverb can make a sound appear as if it is placed inside a physical room. This is done by introducing several phased and decaying echoes in order to simulate wall reflections.

Distortion takes an incoming signal and modifies it to make it sound harsher. A common distortion technique is *waveshaping*, where an input value is mapped to an output value as dictated by a waveshaping function. The simplest (and crudest) form of distortion simply clamps a signal to a certain minimum and maximum amplitude value.

Wave Folding is a waveshaping effects that is based on frequency modulating a 0Hz carrier waveform. This *folds* the input with the waveform and creates harmonic overtones.

Chorus (& Flange) is an effect that fattens a sound by layering several detuned versions of the same sound. It is usually done using delay with a periodically modulated delay time.

Pitch-shifting changes the pitch of an incoming sound without changing the speed of the sound, again simulated using delay.

Phasing is a type of filtering that alters the phases of the different frequency components in the signal using so called allpass filters. A typical phaser modulates these phase shifts, creating an interesting and characteristic swooshing sound.

Compression / **Limiting** is a family of *dynamic processing* effects that alters the input's dynamic range, typically by attenuating amplitude levels when they cross a certain threshold. For compression, the signal can then be boosted, making details in the sound clearer, but without clipping.

Analog vs. Digital

Digital synthesis is in many ways superior to analog synthesis. It is free from static noise and easier to control. However, digital audio has one drawback called *Aliasing*: The Achilles heel of digital synthesis.

Aliasing is a type of distortion introduced by trying to represent frequencies above or equal to half the sample rate: The so called *Nyquist frequency*. Frequencies above that frequency will wrap around the Nyquist frequency and introduce ordered noise in the high frequencies.

Aliasing is usually unwanted, and KarmaFX Synth goes through great lengths to avoid, remove and reduce aliasing by using techniques such as high quality interpolation, band-limiting and oversampling. Still, in some cases, particularly in sample-based and extreme FM/PM synthesis, the Synth cannot avoid aliasing completely. For instance, if you sample a harmonically rich sound played at a low key and play it back 5 octaves higher, aliasing will be heard. How bad it sounds depends on the sample, the pitch, the interpolation technique, etc.

Standard CD quality runs at 44100 Hz. This gives a Nyquist Frequency of 22050 Hz, which all signal frequencies should be below to avoid aliasing.

To reduce aliasing one option is of course to increase the host's sampling rate (e.g., to 96kHz) at the cost of extra CPU cycles, but in general you should not really worry about aliasing. For the majority of sounds this is usually not a problem. Besides, you can force aliasing to come out of almost every digital hardware synth on the market, even the really expensive ones!

MIDI

MIDI (Musical Instrument Digital Interface) is a standardized protocol used for communication between keyboards, MIDI controllers, computers, synthesizers, and even application hosts and VSTs.

Typically, MIDI is used to send *note on* and *note off* events when a note key is pressed and released, respectively on a MIDI keyboard. Data, such as what key (*note index*), how hard it is pressed (*velocity*) is also sent.

MIDI Keyboard/Controller
(Novation Remote SL)

Likewise, control data can be transmitted so parameters can be changed during a song or live performance either using a synthesizer or dedicated MIDI controller. Midi control signals are sent as a sequence of events, where each event carries a value between 0 and 127. MIDI events are usually referred to as *MIDI Control Changes* or simply *MIDI CC*'s.

MIDI Controller
(Behringer BCF2000)

Typical MIDI CC's include *Pitch bend*, and *After touch*: Pitch bend events are usually sent from a MIDI keyboard or instrument in response to changes in position of the pitch bend wheel. After touch tells how much pressure is applied to a key while its being held. Many other MIDI CC's exists (referenced by their hex value) and can be mapped to control user-defined parameters.

4.3 Synthesis in KarmaFX Synth

KarmaFX Synth is built on the same principles as Subtractive-, Sample-, and Additive-synthesis, plus of course Modulation synthesis due to its modular nature, making it a capable and flexible synth and effect unit.

A Monophonic patch only plays one voice while a Polyphonic patch can play multiple simultaneous voices.

Like most other synths, a defined sound in KarmaFX Synth is called a *patch*. A Patch can be played either *monophonically* or *polyphonically*, but only one patch can play at a time in one instance of the Synth. However, the number of simultaneous synth instances is only limited by your systems CPU and memory resources.

Each patch consists of a set of connected modules. Just like a real modular, a module is a single component that either generates or processes a signal. A module has user adjustable parameters, i.e., knobs and sliders, that control its functionality. Almost all parameters can be controlled (modulated) by signals from other modules. We will refer to these signals as *modulator signals* or simply *modulation signals*. Each module can also read and react to MIDI events.

Most modules can work in either mono or stereo mode. If a stereo signal is sent to a mono module, only the left channel of the signal is processed. However, if a mono signal is sent to a stereo module, the mono signal is duplicated to both stereo channels.

Because modules solve different tasks, they are placed into one of these basic categories: Generator, Filter, Amplifier, Effect, Controller, Modulator, or Output.

Generator: A generator creates a sound signal at a certain frequency and hence does not take any inputs. A typical generator is an oscillator producing, e.g., a sawtooth or a square wave.

Filter: A filter processes the input, in the spirit of subtractive synthesis, by removing or boosting certain frequencies. A typical filter is the familiar resonant lowpass filter with simple cutoff and resonance control, known from analog synthesizers.

Amplifier: An amplifier module simply changes the volume (loudness) of any incoming signal, usually triggered by MIDI note-on and note-off events. A typical amplifier is a simple *gate* that turns the volume up when a key is pressed and down when released.

Controller: A controllers has three main tasks. First, it creates control signals (frequency/phase/note/trigger) that are sent to every module feeding the controller. Filters can, e.g., use the frequency control signal for keyboard tracking while oscillators use it to change pitch. Secondly, the controller controls the amount of polyphony, i.e., how many voices the patch is capable of playing. All incoming voices are mixed before they are output from a controller. Finally, controllers can write and alter MIDI events. In polyphonic mode, a frequency control signal is generated for each voice and the controller determines which MIDI events should go to which voice.

Effect: Effects simply process the input in some way. A typical effect is, e.g., delay and reverb. Since effects rarely need polyphony (and in some cases are quite CPU heavy), they are most often placed last in the signal chain.

Modulator: Modulators create or process signals that are meant for controlling parameters, i.e., instead of routing a modulator signal into the sound signal chain, it is wired into a knob to control its value over time. A typical modulator is a Low Frequency Oscillator (LFO) outputting a sine wave.

Output: The output module is a special module that simply sends its given input to the host application. MIDI communication from the host also passes through this module.

The use of modulation signals is what makes modulars so powerful. Any patch cable's voltage can cause changes to one or more parameters of a module. The same basic principle applies in KarmaFX Synth, but some signals are handled somewhat differently: As previously mentioned, the *frequency control signal* is traditionally sent from a keyboard to individual modules (along with a trigger/gate signal to tell when a key is pressed and released).

In KarmaFX Synth this signal still exists, but is created by a Controller module based on received MIDI events. The frequency control signal is then passed on automatically to all modules that transmit their output to that Controller. Triggering is handled by passing on events to busses of MIDI events, which internally generates a *trigger control signal* and *note control signal*. All modules can read and react to the flow of MIDI events on these busses. Aside from the frequency control signal, Controllers can also generate a *phase control signal*.

The neat thing about this design is that you have less control signals to worry about. For instance, typically a keyboard *control voltage* (CV) would have to be wired into many different modules. In KarmaFX Synth this is done for you, by means of the internal frequency control signal, "behind the scenes" so to speak.

New modules that are introduced in Version 2 are shown with a:

NEW IN V.2

tag in the Modules Guide, page 48.

5. User Interface

In order to operate KarmaFX Synth, it is essential to understand its interface. When the synth loads up you'll be presented with a modular environment built up of small, boxed, modules placed on a Workspace and connected by wires.

Modules can be dragged around and connected arbitrarily using the mouse. Each module carries out a task, either generating or processing a sound signal that ultimately is sent to an output module and received by the host.

Each connection is shown as a wire with a small arrow indicating which direction the signal flows. A module has user adjustable parameters, i.e., knobs and sliders, that control its functionality. Almost all parameters can be controlled (modulated) by signals from other modules.

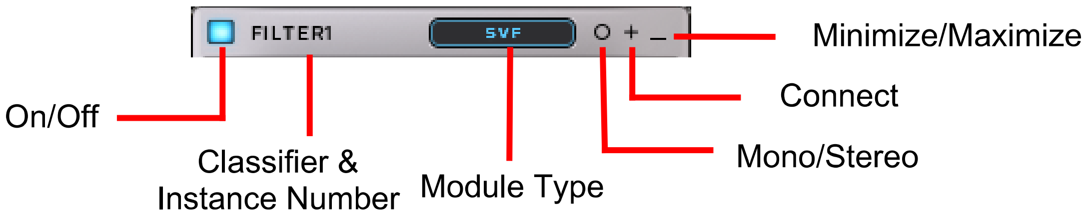
Modules generating signals that modulate parameters (Modulator modules) are directly connected to the particular parameter’s knob or slider. In the bottom of the Workspace is a Control Panel, useful for quickly editing common parameters as well as for browsing and managing patches.

The following sections, describe how to use the modules, the wires, the Control Panel and the menus.

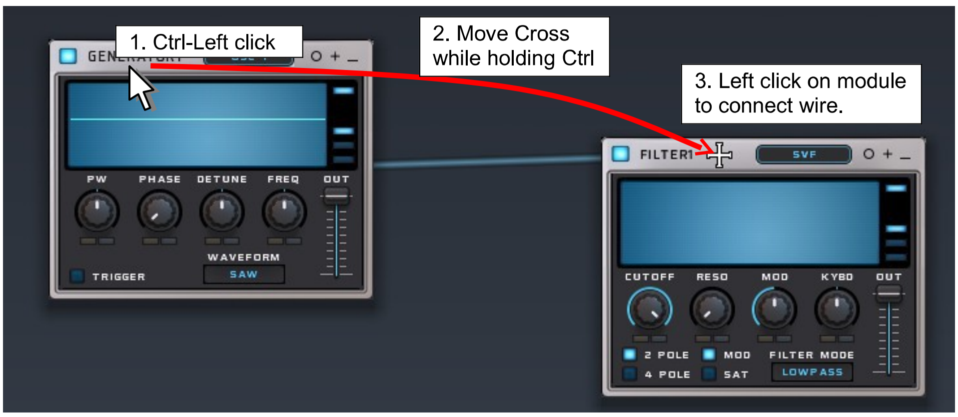
5.1 Modules

A module consists of a *title bar*, a *display area* and a *parameter area*. Some modules also have a special user-interaction display for, e.g., editing envelopes or to display samples. The details of how to operate these is covered in section 7.

Title bar

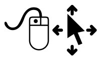
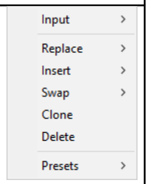
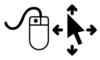
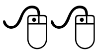


The first thing you notice on the title bar is the *module classifier*, e.g. *Generator, Filter, Modulator*, etc., and its current *instance number*. A dropdown menu chooses the *module type*, i.e., the functionality of the module. If it's a filter module, the dropdown will show a selection of available filters. You can use this selection to quickly try out different types, without breaking the wiring. Audio modules capable of stereo processing have a *Mono/Stereo* switch. A *Connect* button lets you connect one module to another module or to any knob or slider for modulation. A *Minimize* button collapses the module so it only takes up a small part of the screen. Finally, an on/off LED switch can be used to switch a module on or off. When the LED it lit, the module is on. When a module is off it uses no CPU and has no output. Connecting modules is shown below:



Connecting two modules using the mouse: Either hold Ctrl and click the titlebar or click the *Connect* button. Then click on the target module.

You can have up to 16 simultaneous instances of each module class, i.e., sixteen generators, sixteen filters, sixteen controllers, and so on. The only exception is the Output module: There can only be one Output module.

Title bar - Actions	
	**Pressing left mouse on title bar + Drag** Lets you move the module around on the screen. A module automatically snaps to an invisible grid to make it easier to align with other modules. The actual position of the module itself doesn't affect the sound or routing in any way. You should simply try to set it up so it is pleasant to work with.
	**Right click on title bar** Brings up a menu allowing you to select the active *Input* to this module, *Replace* the module with a different kind, *Insert* another module in-front of this, *Swap* the module backwards or forwards in the signal chain, *Clone* the module (and its parameters), *Delete* the module, or save the modules parameters as *Presets* for later retrieval. 
Ctrl 	**Hold Ctrl, click title bar and then on the target module** Press and hold **Ctrl** and click once on the title bar. A small cross appears. Then, while still holding **Ctrl**, click on the title bar or the display of the module you wish to connect to. A wire will appear between the modules to show connection. Alternatively, instead of holding Ctrl, simply click the Connect button.
	**Double click on title bar** Minimize or maximize module (same as clicking on the minimize/maximize icon).

Display

The displays main purpose is to show the final waveform produced by the module. Have you ever sat in front of a silent hardware or software synth and wondered where the sound went, just to realize that the amplifier or the filter was switched off? Then you already know that problems like these are both annoying and time consuming. Moreover, they are even more likely to happen in a modular environment when a lot of stuff is going on. Because of this, KarmaFX Synth offers plenty of visual feedback to keep you in control at all times.

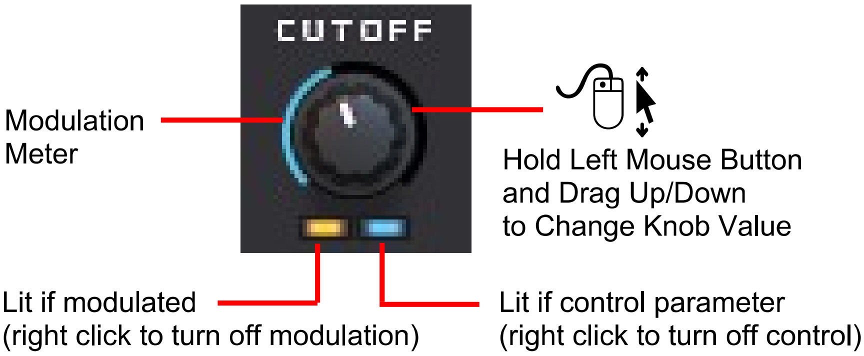
The display is controlled by the four switch buttons to the right of the display. The text area just below the waveform display shows info related to the current module, and parameter values whenever they are changed.

	The first button turns the display on or off.
	Second button switches to time domain display, showing the waveform like an oscilloscope. (This is the default)
	Third button switches to show the frequency spectrum.
	Fourth button turns the display into a special modulation mode, showing the current modulation of the last edited parameter in the module. In this mode, clicking on any of the modules knobs/sliders will show the modulation of that parameter.

Modules running in polyphonic mode will only display the first generated voice in the display. This means that you may hear sound that you cannot see. To alleviate this you can temporarily switch polyphonic controllers to mono mode, observe the generated sound, and then switch it back when done tweaking.

Parameters: Knobs, Sliders and LEDs

Parameters are the most important interaction components in the synth. As on hardware synthesizers they allow you to change and fine-tune a module’s settings – thus controlling the final sound of the patch. To change a knob or slider, simply press the mouse button and move the mouse up or down; or use the mouse’ scroll wheel for fine tuning.

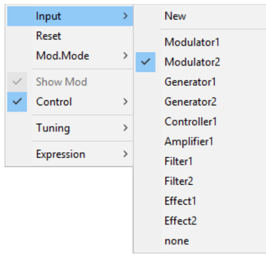


Tip: When changing a parameter, moving the mouse slowly will give you greater precision. Holding the Shift key will also increase precision.

As a special feature, parameters with knobs have two indicator LEDs: The left LED is lit if the parameter is modulated. The right LED is lit if it’s a control parameter.

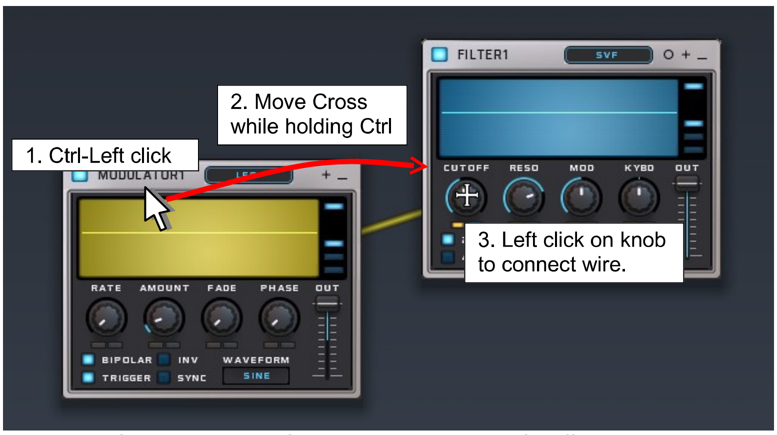
Tip: Holding Ctrl and left clicking on a knob or slider resets its value to center-position.

Almost all parameters in the synth can be modulated. In this context “modulation” simply means controlling a parameter by a signal from another module. To modulate a parameter, simply route the modulator module directly to the knob or slider you wish to modulate. The *Modulation Meter* shows the current modulation value.

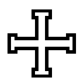


To connect a module to a parameter either right click on the knob/slider and select the source module from the Input menu. Alternatively, either press and hold **Ctrl** and click on the title bar, or on the title bar's Connect button on the source module, and then click on the knob/slider you want to modulate. In any case, a modulation wire will appear showing the connection.

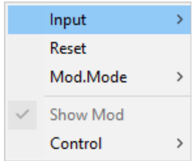
Tip: Choosing *New* in the Input menus adds a new module and automatically patches it to the respective module or knob/slider.



Connecting an LFO modulator to the Filter Cutoff knob


Connection cursor

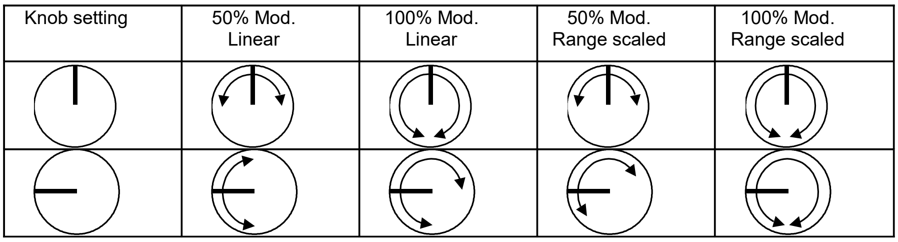
If more than one module is connected to a parameter, their signals are internally mixed before modulating its value.



Besides the Input submenu, the right click menu has some interesting options:

Reset: Resets the parameter value to its default value.

Mod. Mode: Choose between *Linear* or *Range scaled* (default) modulation mode. In linear mode the modulated signal is simply added to the initial setting. In range-scaled mode, the signal is still added, but first after it is multiplied by a scale factor. The scale factor is adjusted so a full range modulation (-1 to 1) will always use the full range of the knob that is modulated.



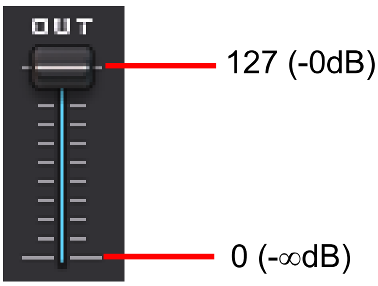
Range scale mode usually sounds "better" because it uses the full range and avoids modulation clamping. Linear mode is best used for parameters that have to be linearly modulated. E.g., range scale modulating a "note" or "octave" parameter can make it go out of tune.

Show Mod.: For Skins that don't have modulation meters, this shows the current modulation of the parameter directly in the interface, by animating the knob or slider. This is very useful to get an idea on what is really going on. However, doing it on many knobs will most likely make your head spin.

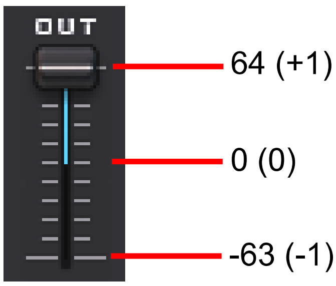
Control: Link the parameter to one of 128 available assignable *control parameters*. A control parameter is simply a parameter seen from the host, meaning that it is possible to change and automate it from within the host application. 32 of the available 128 control parameters appear on the Control Panel in the synth for instant tweaking within the synth (see section 5.4).

Some parameters also have extended options to, e.g., change the parameter’s tuning, range or to make it snap to certain intervals.

Finally, all modules have an output slider for controlling the final amplification of the signal:



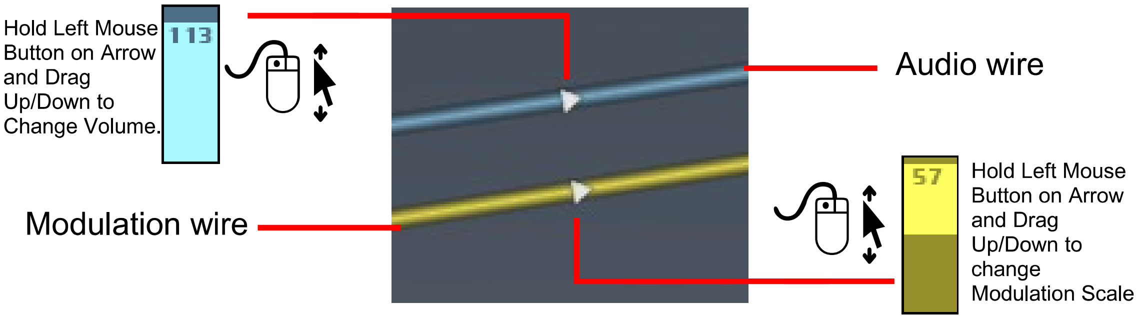
Audio Output slider



Modulator Output slider

5.2 Wires

The wires connect modules to modules (audio wire) and modules to parameters (modulation wire). Thus, there are two kinds of wires, shown in different colors. A small arrow indicates the direction of the signal flow.



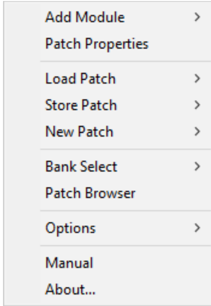
As the above figure shows you can alter the amplification of the signal sent through the wire using the mouse on the arrow head. Right clicking on the arrow cuts the wire and removes the connection.

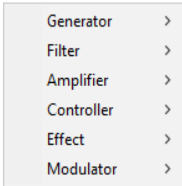
Wire Arrow - Actions	
	Pressing left mouse on wire arrow + Drag up/down Change the amplification of the signal. For audio wires the amplification range is 0 to 127 [0..1]. For modulation wires it is –64 to 64 [-1..1], making it possible to invert the signal.
	Right click on wire arrow Cuts the wire and removes the connection.
or	**Shift Left Click or Double click on wire arrow** Insert module on wire. First, the *Add Module* menu is shown. The selected module is then inserted on this wire, preserving all connections.

The synth does not distinguish between audio and modulation signals. They are treated exactly the same and can be used interchangeably like in a true analog modular system. The wires are only colored differently to make it easier for you to see what's going on.

This means that you can easily modulate a knob or slider using another audio signal, or - if you choose - send a modulation signal (e.g., from an LFO) into an audio module.

5.3 The Right Click Menu

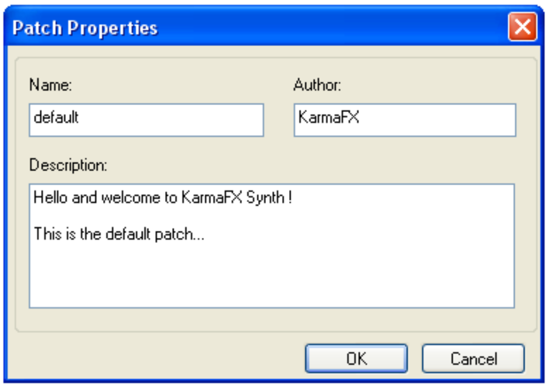




The main right click menu gives you easy access to common patch editing tasks, like adding new modules, loading and storing patches as well as changing general options in the synth. To open it, simply right click on the workspace background.

Add Module: Creates a new module of your choice selected from the submenu. The chosen module is placed at the mouse cursor.

Patch Properties: Shows the patch properties dialog box. Here you can change the name of the patch and the name of the author. If you want you can also add a description of the patch.



Load Patch: Directly choose a patch to load from the current patch bank. A bank is simply a collection of up to 128 patches stored in one folder.

Store Patch: If a bank contains vacant slots, these will also be select-able (unlike the load patch submenu). You can also select an existing patch if you want to replace and overwrite an already existing patch. The synth will ask for a confirmation before overwriting any patches.

Bank Select: Select the patch bank to browse from in this submenu. Note that this selection will not alter the currently loaded or edited patch. This means that it is possible to copy a patch from one bank to another, by first loading the patch from one bank, then selecting another bank, and finally storing the patch in that bank. Selecting a *<New Bank>* item will prompt for a new bank name, and when OK is pressed, creates a new bank folder by that name and switches to that bank. Use this when you want to roll your own patch banks.

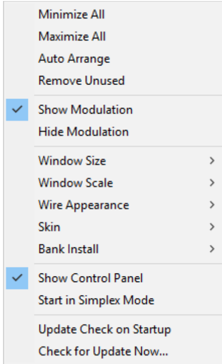
Patch Browser: Opens the Patch Browser (section 5.6). Can also be achieved by clicking on the Control Panels display (section 5.5).

Options: Shows the options submenu. Here you can alter various overall settings in the synth (see section 5.4).

Manual: Opens this manual in PDF format.

About: Shows the About Dialog for KarmaFX Synth Modular. Here you can see version information, the current build number and date, as well as the name of the registered user (hey, that's you! ☺).

5.4 The Options Menu



The Options menu gives access to overall settings in the synth:

Minimize/Maximize All: Minimizes (collapses) or Maximizes (expands) all modules in a patch. The minimize mode is useful for getting a quick overview of the wiring in the patch, since everything but the title bar of the modules is hidden, leaving good room for the wires. Modules are by default always maximized.

Auto Arrange: Relocates all connected modules into a grid starting from top left corner and moving across the screen. This can sometimes help clean-up a messy patch where modules have been moved so they either overlap or aren't aligned with the grid.

Remove Unused: Deletes modules that are either disabled or not connected, while retaining all existing wiring. This is a quick way to clean-up a patch that contains many inactive modules, while ensuring that it sounds exactly the same.

Show/Hide Modulation: Shows or hides modulation of all modulated parameters. When *Show* is enabled, knobs and sliders are animated to show what effect the modulation has. This is the same as right clicking on each parameter and selecting "Show Mod.". It is useful when creating new patches, to verify that parameters behave as expected when modulated. Default mode is to hide all modulation. This option is now superseded by the built-in knob metering in the version 2 skins, but is still available for backwards compatibility with the classic skins.

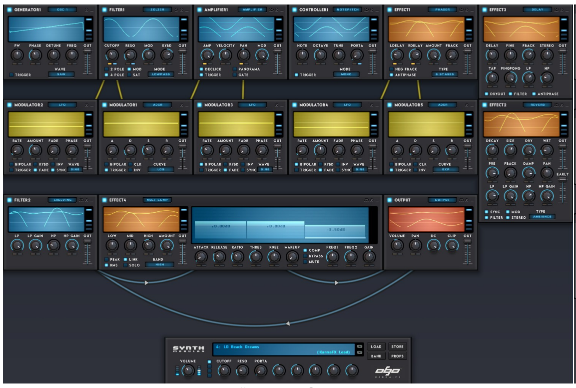
Window Size: Choose between 1500x1000 (default) or custom resolutions (up to 4096x2048; minimum size is 400x400). Low resolutions are useful when running on laptops with limited screen size. The window size should change instantly. If not, please close and reopen the plugin. Patches saved in custom resolutions higher than the default might appear differently when reloaded in the default resolution, as modules may have to be relocated to make them fit on screen.

Window Scale: Scale the UI by 100% (default) or 200%. On skins that support it, extra high-quality scales are available: 125%, 150%, 175% 200%. This is useful when running on a system with a high resolution screen setting, such as Retina or similar high DPI resolutions. The window size should change instantly (and if not, on reopen). On Mac/OSX Retina scaling should be manually activated using the HiDPI/Retina setting.

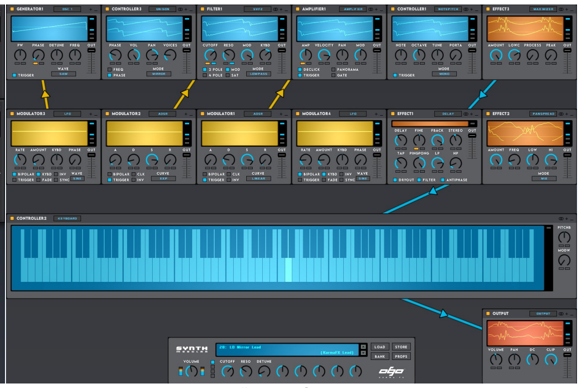
Wire Appearance: This changes the way the wires are drawn. Choose between Linear (default), Curved or Dangling. This is basically just eye-candy and a matter of taste. Just choose what suits you best.

Linear Curved Dangling

Skin: Changes the skin, i.e., the graphics used to draw the user interface (modules, knobs, slider, wires, etc.). Again a matter of taste. The "Ivory" skin is used for the screenshots in this manual. Here are some alternative skin examples:

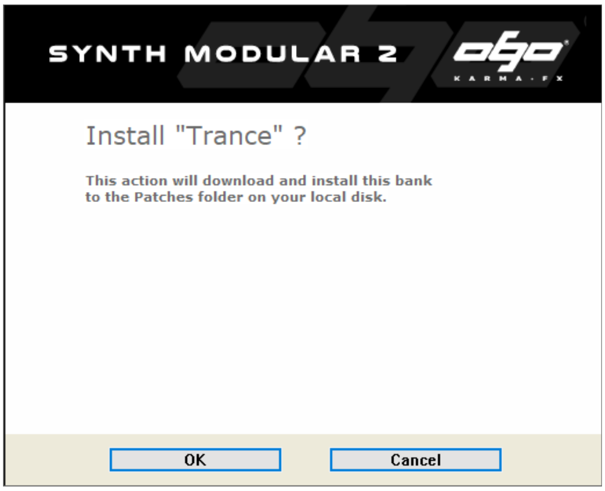


"Dark" Skin

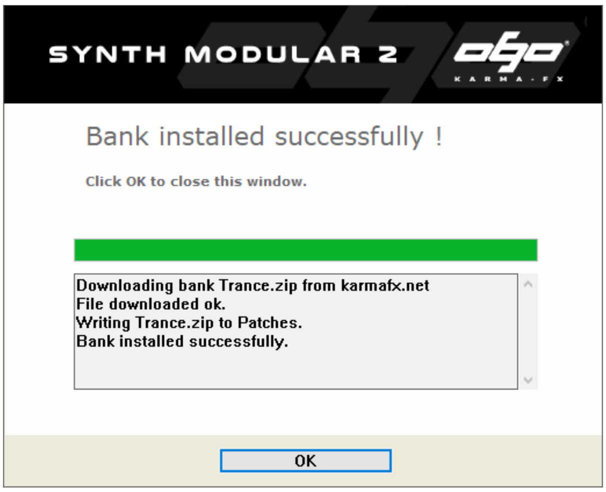


"Neo" Skin

Bank Install: This menu lists the patch banks that are available online. Since KarmaFX Synth is created to work offline as well as online, you need to select the **Update Bank List** menu item, for the synth to contact the server and download the latest patch bank list. Patch banks that have already been installed are shown grayed out. Clicking on a bank will ask if you wish to install this bank to your local disk.



Click **OK**, and the synth will proceed to download and save the chosen patch bank in the Patches folder. Hence, the downloaded patch bank is made available instantly for you to browse and use.



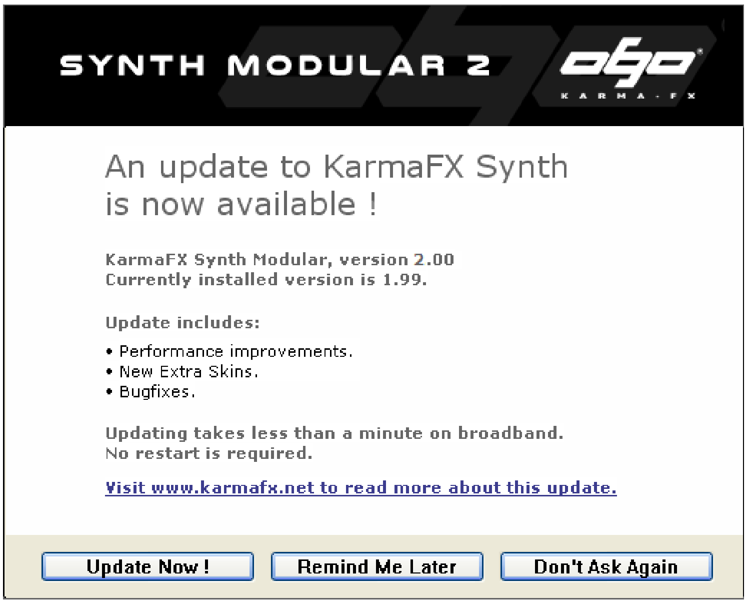
Show Control Panel: When enabled (default and recommended setting) a Control Panel is shown in the bottom of the screen. How to operate the Control Panel is covered in section 5.5.

Start in Simplex Mode: When enabled, new instances of the synth will start in *simplex mode*. Simplex mode is a special patch editing mode where the interface is shrunk so only the Control Panel is shown. This of course takes up a lot less screen space but also limits patch interaction considerably. You can toggle simplex mode on and off manually by clicking on the Synth modular logo in the Control Panel.

Synth running in simplex mode

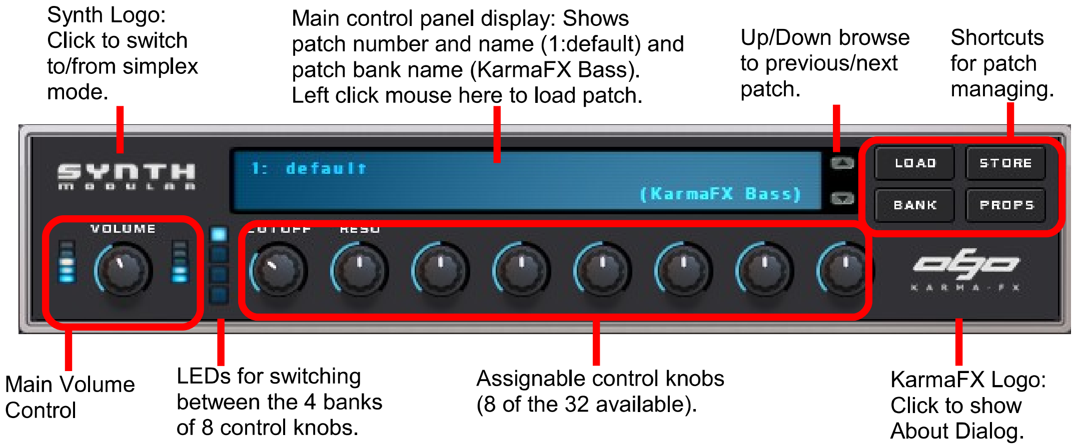
Update Check on Startup: When enabled and internet access is available, the synth will check for new updates on startup.

Check for Update Now: Use this to check for updates manually. When a new update is found, the version information is shown and the updates main new features are listed, and you'll be prompted whether to install it or not. Updates can require the system to download and launch the full installer, so the host application will have to be closed down and restarted once the installation completes.



5.5 The Control Panel

The Synth's Control Panel located in the bottom of the workspace is useful for editing and browsing patches. The main display shows the current patch name and bank name. The arrow-buttons to the right of the display are used for quickly browsing to the previous/next patch.



The four rightmost buttons are shortcuts to **Load** and **Store** patches, change **Bank** or properties (**Props**). The Load Patch button brings up the Patch Browser (See Section 5.6.), while the rest work exactly as when choosing the same options from within the right-click menu,


Load Patch


Store Patch


Bank Select

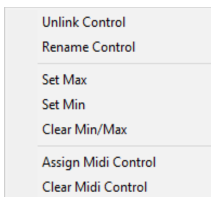

Patch Properties

The leftmost knob controls the **Volume** for the patch. This knob is simply linked to the volume knob on the Output module. The LED's to both sides of the knob indicate the current output level in the left and right channels respectively. It is good practice to make sure that it doesn't reach the red level, but the synth will not clip the signal (unless you actively turn on clipping in the output module). So even if it's in the red, you can still pull down the volume in your host application a get a clean (unclipped) sound.

The row of eight knobs below the main display, give direct access the user-linkable controls, in 4 sets of 8 knobs. The LED's to the left, switch between the 4 sets of knobs. Whenever a knob or slider is linked to a control it will automatically appear here, as well as inside the host application. This means that you can tweak it and even record it in the host for automating parameters in a song. If you want, several different parameters can be linked to the same control knob.

Control knobs are assigned by right clicking on a knob and choose *Control->*, and should be used on parameters that you want to automate.

Right clicking on a Control Knob will bring up a menu, giving you the option to **Rename Control** to a different name or **Unlink Control**. The Control Knob's ranges can be reprogrammed by using the **Set Max** and **Set Min** options, and reset again using **Clear Min/Max**. You also have the option to assign a knob to a specific MIDI Control (MIDI CC) for easy editing of parameters from e.g. an external MIDI device. Click **Assign Midi Control** to start *MIDI-Learn* mode. Then change a knob or slider on your MIDI device to automatically assign that MIDI CC to the control knob. For this to work, make sure the host transmits all MIDI events to the plugin. Use the **Clear Midi Control** option to disconnect the MIDI Control again.



5.6 The Patch Browser

The Patch Browser offers a way to quickly browse patches and patch banks. The left side of the Patch Browser window is the *bank view* that shows the available banks, while the *patch view* on the right side shows all the patches in the currently selected bank.

You can switch bank by simply left-clicking on a bank name in the bank view. To scroll the bank view, you can click on the arrows next to the view or use the mouse-wheel while hovering.

To select a patch, you can left-click on the listed patches in the patch view, and it will load instantly.

On hosts that support plugin keyboard-input (and while the Patch Browser has focus), you can also use the keyboard's arrow-keys to navigate both patches and banks, or even switch between the bank and patch views. This is useful if you want to quickly browse the patches, while trying to find a particular sound.

Using the keyboard, it is also possible to filter search the patch view. To enable keyboard filtering, click the panel button on the far right. Then type the name of a patch to instantly filter the patch view according to the typed search text. Only the patches that match the text or part of the text will be shown.

The search text is shown in the bottom right of the patch view. To clear the filter, either repeatedly press backspace to remove the text, hit the Escape key, or simply click the panel button again. Alternatively, closing the Patch Browser and opening it again will also clear the filter.

To close the Patch Browser, either click the Load button again, or hit the Escape key. Clicking on the Workspace background will also close the Patch Browser window.

6. Basic Operation

So far we have looked at the Synth's interface and its main features. Now, it's time to start using those features! This section shows how to setup KarmaFX Synth in your host application (either as instrument or effect) and start browsing the built in patches. Once familiar with the basics, we will use our newly gained knowledge from the previous sections to build a simple patch and save it for later retrieval.

6.1 Using the Synth as Instrument and Effect

Before you can start using the Synth, you need to know how to insert and use VST/Audio Unit plugins inside your host application. The way to do this differs from host to host, but normally you'll have the option to drag or select a VST/Audio Unit from a plugin-list and place it in an audio and/or MIDI track.

The Synth can operate in two modes: As instrument (VSTi/Audio Unit Instrument) or effect (VST/Audio Unit Effect).

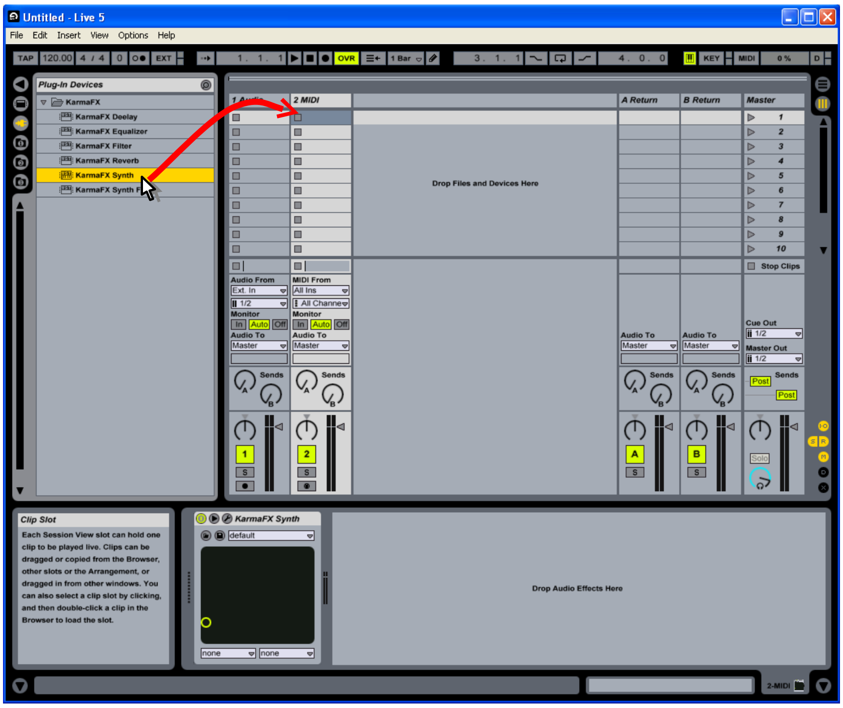
If the synth should generate sound based on the played nodes, it should be used as an instrument. The instrument generates audio based on MIDI events, so it has to be placed in MIDI track.

If the synth should alter the sound output of a track, it should be used as an effect. The effect needs to be put after any audio generators, and can be placed in an audio-, aux- or even master track.

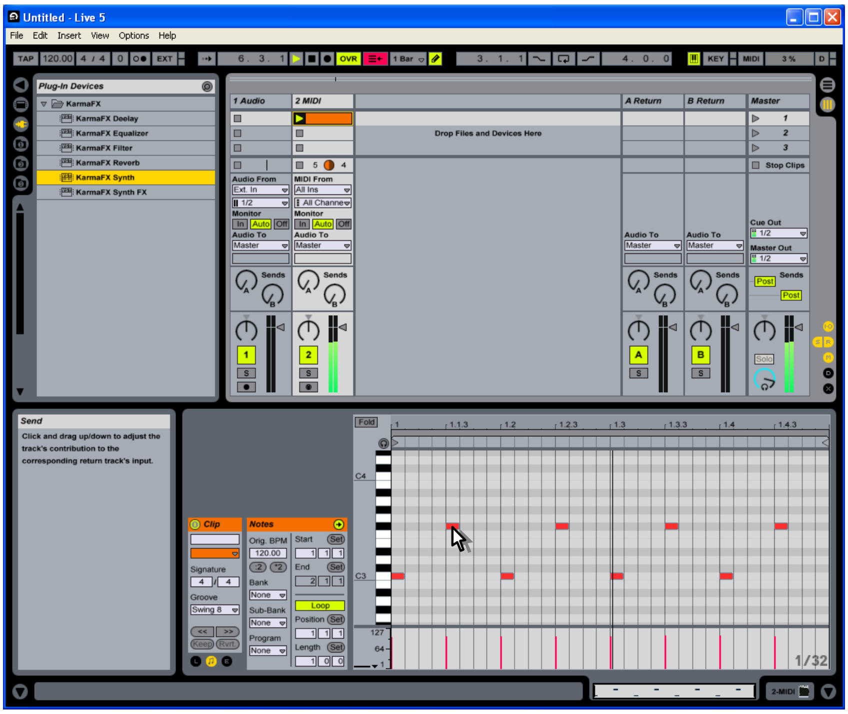
The following illustrates how to use KarmaFX Synth in the host Ableton Live. You host of choice might work differently. If in doubt, please refer to your host applications users manual to learn how to use VST plugins.

Using KarmaFX Synth as Instrument in Ableton Live

▶ Locate the KarmaFX plugin folder in the host's list of VST plugins. Then drag KarmaFX Synth to a vacant MIDI track.



▶ Enter a sequence pattern and hit play to hear the synth in action. By default the synth starts up with a default bass patch.

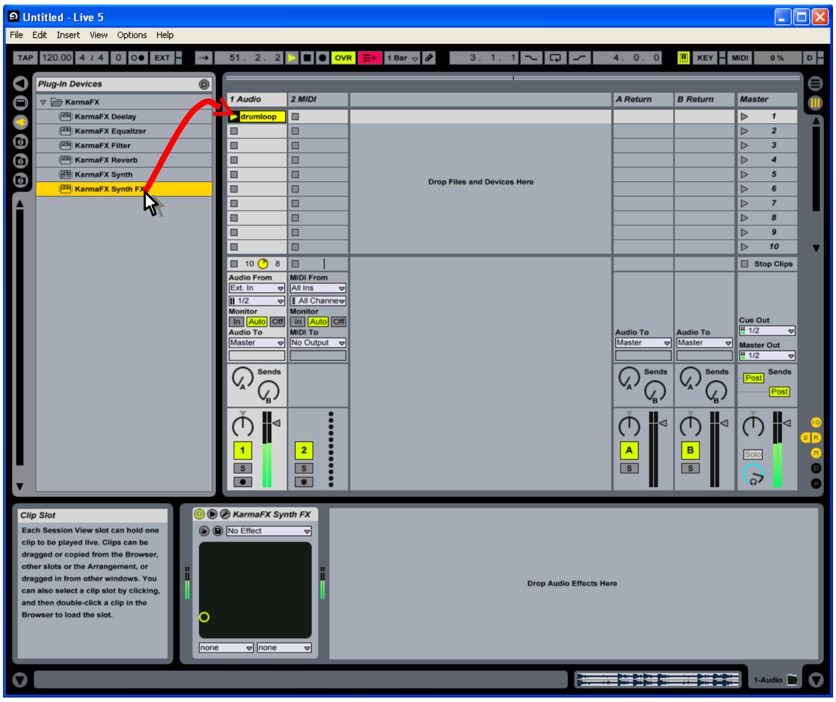


▶ Alternatively, if you have a MIDI keyboard connected, click on MIDI input monitoring, and play a few notes on your keyboard.

Using KarmaFX Synth as Effect in Ableton Live

▶ First, click on the audio track you wish to add the effect to (preferably one that generates sound).

▶ Now locate the KarmaFX Synth FX in the VST plugin list and drag it to the effect area in the bottom of the screen.



▶ By default KarmaFX Synth FX start up with the KarmaFX Effect bank, so you can start using it as an effect immediately. The default effect patch does exactly nothing, but browse down a few patches and you'll find a simple delay. Then tweak a few knobs and hear how it changes the decaying echo.

6.2 Browsing and Handling the Built-In Patches

The synth has more than 1000+ different patches to choose from. Patches are placed in *banks*, where each bank contains up to 128 patches. The banks are arranged and named according to their category. For instance, there is a "Bass" bank, a "Pad" bank a "Drum" bank, and so on.

By default the synth starts up with the "KarmaFX Bass" bank (or "KarmaFX Effect" bank when using the effect plugin). To browse/use and listen to the many patches in a bank there are several options:

▶ Use the Patch Browser to choose bank and patch directly (See section 5.6). To open the Patch Browser, use the Load button in the Control Panel or simply click the Control Panels display.

▶ Use the Control Panel's up/down arrows to go to the previous/next patch.

▶ Right click on the background and select "Load Patch…". Then select a patch from the submenu that appears. Note that, when this is invoked from within a subpatch, the patch is loaded into the subpatch.

▶ Select a patch directly from within your host as a VST preset (The Synth's GUI does not have to be open to change patches this way).

Once a patch is selected, play a few notes on your MIDI keyboard or start to loop a sequence in your host to hear it.

To switch bank choose the "Bank Select" menu item from the right click menu or click the bank button on the Control Panel or use the Patch Browser. Selecting a new bank will not alter the currently loaded or edited patch, so to actually load a patch from the newly chosen bank you have to actually load a patch. Besides not losing any unsaved changes to a patch, this has the advantage that you can copy a patch from one bank to another (see section 5.3).

Once a bank has been selected you can also browse the patches from within your VST host application. The patches will appear as *program presets* in the host. You can even export the selected patch as an .FXP file in most host applications, if you would like to store a patch outside the Synth, e.g., to send a single patch to a friend. You cannot however export an entire bank as .FXB as some host applications permit. To take a backup or share an entire bank with a friend, you should locate the KarmaFX_Synth/Patches folder in the installation folder, and copy the folder with you bank's name.

On disk, a bank consists of a collection of patch files (.kfx) and an index file (index.txt) located in a single folder. Normally you wouldn't have to mess with these files. Still it is nice to know that if a bank's index file is missing, the Synth will attempt to recreate a new index file from the files residing in the folder so that it will load correctly.

Always be sure to backup your homemade patches stored within the built-in banks before installing newer versions of the Synth. Otherwise these patches might be overwritten on installation. To be on the safe side, it is recommended that create a new bank with a unique name and store your own patches there. Then, when re-installing the Synth, your patches should not be overwritten. But it is of course always a good idea make a backup copy - just in case.

6.3 Creating Your First Patch

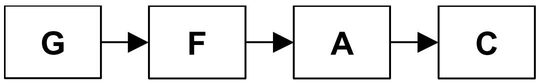
Let's build a patch from scratch! We'll start by making a patch template schematically to make this section as general as possible. In fact we can use this template as a basis to construct almost any future patch.

▸ First, we place a generator (denoted **G**). What kind of generator is not so important. It simply creates a sound at a certain frequency:

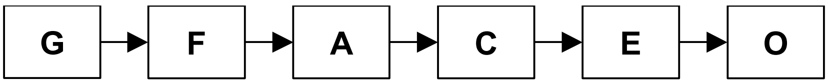
▸ To make it interesting we feed its output into a filter **F**, and send the result into an Amplifier (**A**). For now, let's just say that this is a simple gate that passes sound when a key is pressed:

Notice how this corresponds to the vintage analog VCO➝VCF➝VCA patching**.**

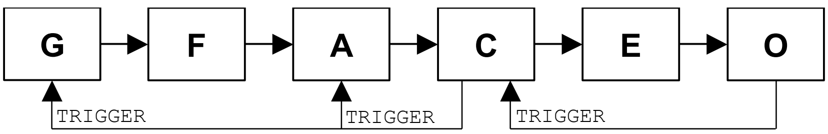
▸ We are now ready to send the results into a controller. The controller will handle all MIDI-triggering of the Generator, Filter and Amplifier as well as send a frequency control signal to these modules:



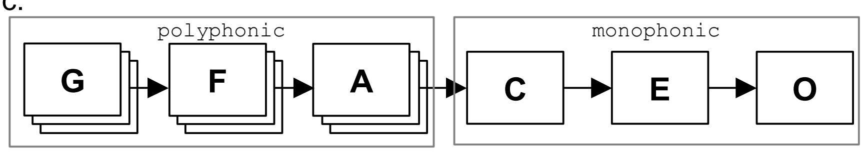
▸ Finally, we send the result into an Effect (**E**) and out to the Output module **O**:



Our simple patch template is basically done. So what would happen if we press a key? Some modules have a Trigger option that tells whether they listen and respond to MIDI events. In our schematic this is true for the Generator, the Amplifier and the Controller. MIDI events sent from the host (through output) trigger the controller, that subsequently passes these events on to the Generator and Amplifier in order to activate them:



If we set the Controller to polyphonic mode, the modules placed before the Controller in the signal chain will behave polyphonically, while the ones after will still be monophonic:

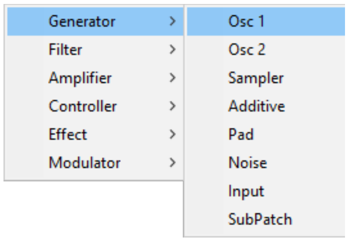


This happens automatically and ensures that only modules that really need multiple voices will get it.

Now let's implement the patch-schematic in the Synth!

▶ Startup an instance of the Synth in your host application. The default bass patch is loaded on startup. Right click on the background and select **New Patch➝Empty Patch**. This clears the workspace, leaving only an output module.

▶ Double click on the background to create a module. We start by creating a simple sawtooth oscillator: **Generator➝Osc1**.



▶ Repeat the same process to create **Filter➝SVF**, **Amplifier➝Amplifier**, **Controller➝NotePitch** and **Effect➝Delay**. Rearrange them as shown by pressing the mouse button on a modules title bars and dragging them.

▶ Now we have to connect the modules. Start by connecting Osc1 to SVF. Either hold the **Ctrl** key, click on Osc1's title bar and then click anywhere on the SVF module (with the exception of any knobs & sliders) - or - click the **Connect** button on Osc1 (the small "cross" icon) and then click on the SVF module.

▶ Do the same for the other modules. You can wire several modules in a row by continuing to hold the **Ctrl** key.

We're done! First, check that the wiring and the arrows on the wires are as shown. Then try to play a note in the host's sequencer or on your MIDI keyboard. You should hear a bright sawtooth buzzing sound with a bit of echo. Not very impressive, but it's a start.

Let's dig in to the details of how and why it works: By default Osc1 generates a sawtooth wave at a pitch chosen by the controller. The trigger option on Osc1 selects whether the waveform should phase restart on a note-on MIDI event.

The Amplifier is by default set to *Gate* mode, meaning that it only lets the incoming signal pass between a note on and note off MIDI event pairs. The Controller is set to mono mode, i.e., only one voice is allowed to play at a time.

Try turning the cutoff knob on the SVF module. Since it is set to lowpass by default you should hear the familiar attenuation of high frequencies/harmonics.

▶ We can easily make this patch more interesting. Let's try setting Osc1's waveform to a square wave. Now create an LFO modulator and route that into Osc1's pulsewidth knob. You do this much like when routing a module to another: Either hold the **Ctrl** key, click on the LFO's title bar, and then, while still holding **Ctrl**, click directly on the Pulsewith knob - or - click the **Connect** button on LFO's title bar and then click the Puslewidth knob.

► Set the LFO's amount to around 50 and its rate to 42 (or 0.5 Hz). This modulates the pulsewith of the squarewave over time to fatten up the sound.

► Now create an ADSR modulator and route that into the Amplifier's Amp knob. Turn off **Gate** mode to let the ADSR fully control the sounds amplitude. Now try to hit a key and listen to how you can shape the sound using the Attack, Decay, Sustain and Release knobs.

As you have hopefully experienced it's really easy to start making some basic patches and experiment by adding and connecting different modules. Please refer to the module guide in the back of this manual for explanations of more modules. And feel free to go crazy and experiment!

6.4 Saving Patches

By default patches are saved within the song in the host application. This means that if you load up a patch, make some changes and then save the song. The patch is stored too and will reappear exactly as when you saved it the next time you load the same song.

You may also choose to save a patch in a bank. This is useful when using the same patch in different songs, or when rolling your own patch banks. To save a patch, right click on the background and select "Store Patch" or click on the "Store" button on the Control Panel. Then select the patch slot you wish to store your patch in. If the selected slot already contains a patch, the system will ask if you wish to overwrite it.

Some Patch Banks may be *read-only*, which means that you can load but not save patches to the bank. To save a patch you instead have to create a new bank or select a patch bank that is not read-only. Internally a read-only bank is simply a ZIP compressed folder, so another way to make it writable is simply to unzip it to the patch bank folder.

Creating you own patch bank from within the synth is easy too: Simply choose "Create bank…" in the main right click menu.

There are several reasons to why you might want to make you own patch bank. First it enables you to store your work in one place, so you don't have to worry about remembering in which song you used your patch. It is also easier to distribute patches, if you should feel generous and wish to share your work with others. Finally, your work can easily be backed up and restored if you have to reinstall the synth or move the installation to a new machine.

Once you are done with your own patch bank, you can compress the entire patch folder using ZIP compression, making sure to name the compressed file the same as the original patch bank folder - just with an added ".zip" extension. The synth will then read patches directly from a compressed bank, but - as mentioned - it cannot write to it. So aside from saving disk space this effectively makes your patch bank read-only as well as easy to distribute.

6.5 Working with Subpatches

The synth features a **SubPatch** module available as both Generator and Effect (section 7, p.102). This is an advanced feature. Unlike any other module, a SubPatch module can contain 1 to 8 patches (called *subpatches*). This allows for patch construction or loading inside an existing patch, and the option to easily switch between these (sub)patches.

The knobs on the SubPatch module are programmable, and map directly to the control parameters of the currently selected subpatch. This means that you can operate, automate and even modulate parameters in the subpatch from the *parent* patch.

The SubPatch module features and **Edit** mode, that allows you to Edit, Add, Remove, Clear, Load and Rename subpatches. Edit will open the subpatch for editing, meaning that the existing (parent) patch construction is pushed aside, and the interface is changed to show the subpatch construction. After editing, you return to the parent patch by clicking **Exit SubPatch** on the control panel. While editing, the parent patch is still loaded and active, meaning that if you play MIDI, you will always hear the sound produced by the topmost parent patch.

The simplest way to create a subpatch is to add a SubPatch module and click Edit. You can then add and wire up modules, like with any other patch, and choose Exit SubPatch when done. A Generator SubPatch is designed for subpatches that don't require any input, such as Oscillators or even entire instruments. The Effect SubPatch however is designed for effect subpatches that do require input. In those patches, the Input module features the mixed signal of all the inputs going in to the SubPatch module. Hence, you can consider the Input module the entry point, and the Output module the exit point in any subpatch. It is important to note that subpatches are unique and saved with the topmost patch, and are not simply “links” to an existing patch on disk.

You can both load and save subpatches separately from the parent patch. The simplest way to load a subpatch is to choose Load from the SubPatch module. This bring up the Patch Browser, and allows you to select a patch from any bank, which is loaded directly into the currently selected subpatch slot. An alternative option is Edit the subpatch, and then choose “Load Patch” from the workspace's right click menu to load a patch from the currently selected patch bank. A warning is displayed, to ensure that you are aware that you are loading into a subpatch. Similarly, choosing “Save Patch” allows you to save the subpatch. Again, a warning is displayed to ensure that you are aware that you are not saving the topmost parent patch, but just the subpatch.

Subpatches are a powerful tool for simply tidying up and organizing patches, or for creating more complex patch constructions. For instance, you can easily build a patch containing your favorite chain of effect modules, save it as a separate patch, and then load this patch into instrument-patches containing an Effect SubPatch, thereby reusing and applying the same effect to a whole series of patches. Or consider a case where you like a certain patch, but want to modulate its parameters in a creative way. Then you can easily load this patch into a Generator SubPatch module, and hook up LFO's or other modulation sources to its control parameters, and save the result as a new unique patch. The synth features up to two nested levels of subpatches, allowing you to load subpatches that already contain subpatches.

6.6 Patch and Patch Bank Creation Guidelines

If you would like to contribute to this project and share your great patch creations with the rest of the world, here are a few tips and guidelines for creating good quality patches and patch banks. Now, these are just *guidelines*. Don't take them too seriously. Most of these are simply to make sure that patches are easily accessible and easy to operate. If you don't feel like following all of these, or think your patches are good *as-is*, that is fine too. All patches and patch bank contributions are welcome and much appreciated!

That said, here are some basic patch and patch bank guidelines:

1) A patch should ideally sound "good" by itself and be musically interesting. I.e., imagine a patch that you hear for the first time and instantly want to play.

2) A patch should be tuned in middle *C* in *Concert Pitch* (tuning A=440Hz), and respect **Note-On** & **Note-Off** events.

3) A patch should preferably have at least one parameter mapped to the **Control Panel** for automation.

4) A patch should preferably have **Expression** (e.g., *ModWheel*) setup.

5) Modules should be arranged nicely (use **Auto Arrange** option), so it is easy to follow the signal path and see what is going on.

6) Modules should preferably be contained within the default UI resolution.

7) Unused modules should be removed (use **Remove Unused** option), unless they serve a purpose. Sound samples loaded into deselected modules should be cleared, to not take up space.

8) A patch should be light to gentle on the CPU. This means not going too crazy with Unison or Polyphony, and rearranging modules so all **EQ** & **Effects** are placed <u>after</u> the **Amplifier** and **NotePitch** modules in the signal chain.

9) A patch should use Filters in **Exponential** mode. This is the default. But if you use an older patch as base that might not be the case, and filters may have to be switched manually.

10) Polyphonic patches, Pads etc. should use *ADSR* Envelopes with **Analog Attack** turned off. Again, this is the default, but if you use an older patch as base that might not be the case, and **Analog** may have to be switched off manually.

11) Patches should be named according their type: BA:Bass, LD:Lead, ARP:Arpeggio, PD:Pad, DR:Drum, PL: Plugs, FX:Effect, CH:Chords, SY:Synth. Patches can also be grouped logically with separators, either by manually editing the *index.txt* file in the patch bank folder, or saving a patch named "-".

12) Patches should have more or less equal volume and tuned to not clip on output.

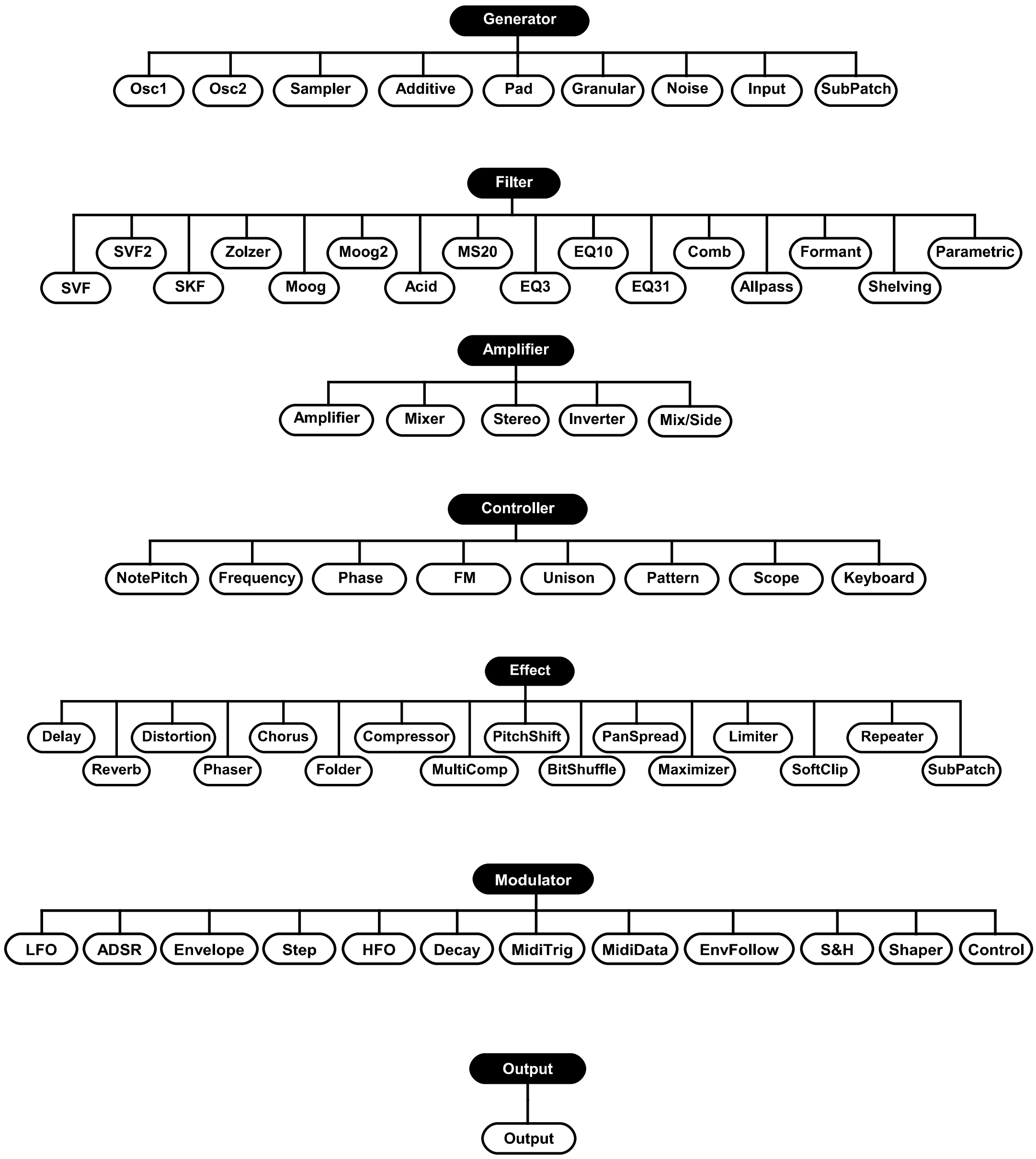
13) Patches should not contain copyrighted samples or other material that isn't royalty-free, unless of course you are the copyright owner.

14) Patch banks should be zipped for easy distribution and installation.

7. Modules – The Complete Guide

This section describes each available module, explaining their functionality in detail, as well as how to operate them. It is suggested that you skim these pages (without digging too much into the details) the first time you read them. Afterwards, you can use this section for reference whenever you feel that a module' or, e.g., a knob's functionality is unclear.

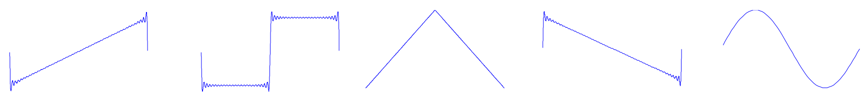
Here is an overview of all the available modules:



Osc1

Generator

Osc1 is a simple oscillator capable of simulating analog waveforms: Saw (sawtooth), Square, Triangle, Ramp (inverted saw) and Sine.

Parameters	
PW	**Pulsewidth** Square and Triangle are capable of pulse-width changes. For all other waveforms this knob has no effect.
PHASE	**Phase** Controls the **Phase** of the oscillators waveform (i.e., where in the waveform are we?), useful for Phase modulation. When Osc1 is triggered, one can optionally have the waveform restart its phase cycle (See Trigger). **Phase** also adjusts the start offset into this cycle. A **Phase Init Only** option is available in the knobs right click menu, that when enabled means that **Phase** only controls the **Phase Init**. A **Random Poly Phase** option is also available, adding a randomized Phase Init offset when Osc1 is triggered polyphonically. Useful for super saws and strings.
DETUNE	**Detune** Adjust the frequency up or down steplessly one semitone.
FREQ	**Frequency** Adjusts the frequency up or down continuously one or more octaves. Octave range and optional snap can be set through the right-click menu.
TRIGGER	**Trigger** When enabled the signal will phase restart on note-on events.
WAVE	**Waveform** Select the waveform to output: Saw, Square, Triangle, Ramp and Sine:  Saw Square Triangle Ramp Sine

Osc2

Generator

Similar to Osc1, Osc2 simulates analog waveforms, but instead of just one oscillator it features two (dual) oscillators. By default the waveforms run at the same frequency, but they can be set to run out of tune, resulting in a "fatter" sound. The second waveform can *Hardsync* to the first, meaning that it is forced to phase restart whenever the first waveform does. This adds an interesting characteristic to the sound, especially when the frequency of waveform 2 is changed. Finally it's possible to *Ring modulate* the two signals, resulting in "robotic"-like sounds.

Parameters	
PW	**Pulsewidth** Square- and triangle-, and sine-waves are capable of pulse-width changes. For all other waveforms this knob has no effect.
PHASE	**Phase** Controls the **Phase** of the oscillators waveform (i.e., where in the waveform are we?), useful for Phase modulation. Osc2 offers the same Phase options as Osc1 through the knobs right click menu: **Phase Init Only** and **Random Poly Phase**.
TUNE	**Tune** Adjust both frequencies up or down continuously one semitone.
FREQ	**Frequency** Adjusts both frequencies up or down continuously one or more octaves. Octave range and optional snap can be set through the right-click menu.
MIX	**Mix** Controls the mix-ratio between the two waveform signals. Rotated left means 100% waveform 1, rotated right means 100% waveform 2. Centered means 50% of both.
RINGMOD	**Ring Modulation** Sets how much ring modulation to mix with the signal. 0 means 0%; 127 means 100% ring modulation.
TUNE2	**Tune 2** Offsets the frequency of waveform 2 up or down one semitone.
FREQ2	**Frequency 2** Offsets the frequency of waveform 2 up one or more octaves. Octave range and optional snap can be set through the right-click menu.
WAVE 1/2	**Waveform 1 / Waveform 2** Select the waveform to use for each oscillator. Choose between: Saw, Square, Triangle, Ramp and Sine. Unlike Osc1, Sine can be pulse-with modulated, since it is a "fake" sinewave derive from the Triangle.
TRIGGER	**Trigger** When enabled the signal will phase restart on note-on events.
HARDSYNC	**Hardsync** When enabled waveform 2 will hardsync to waveform 1.

Sampler

Generator

The Sampler module is a device capable of playing arbitrary waveforms stored as 16, 24 or 32 bit samples. One can control the playing frequency as well as start/end position and optional looping of the sample. To play a sample at the correct pitch the module must know its "key" (frequency of the fundamental harmonic). This is by default set to C2 (65.40 Hz), but can be changed to any valid semitone.

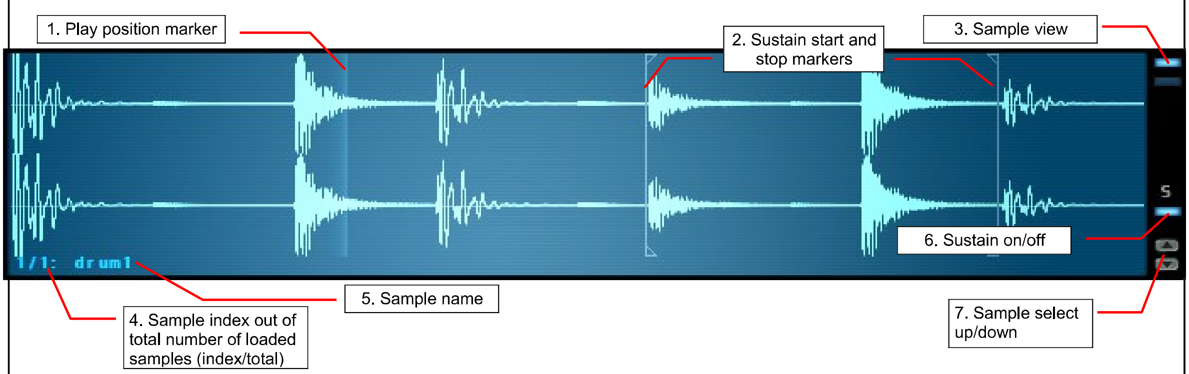
Multiple samples can be loaded into one Sampler, but only one sample can play at a time. Sample switching can be done manually; or automatically by setting key/velocity ranges for each sample (Multi mode). Two types of looping are supported: Sustained looping that is only active when a key is held (set for each individual sample); or global looping between the positions dictated by the start and length parameters. In order to avoid file and disk dependencies, all samples are stored within the patch.

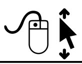
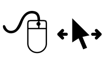
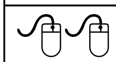
Parameters	
START	**Start Position** Sets the relative start position, where the sample will start to play when triggered and loop to when looped.
LENGTH	**Length** The relative length of the sample range to play and/or loop.
DELTA	**Delta** Adjusts the relative playing speed. Similar to slowing down or speeding up a record on a turntable. A *Tempo Sync* option is available in the right-click menu. When enabled (default is disabled) the playback rate will sync to the current tempo. Useful for, e.g., drum loops.
FREQ	**Frequency** Adjusts both frequencies up or down continuously one or more octaves. Octave range and optional snap can be set through the right-click menu.
TRIGGER	**Trigger** When enabled the sample will start to play on note-on events. When off the sampler will run continuously (free running).
MULTI	**Multi** Turns on multi sampling mode. In this mode the module uses the assigned key/velocity ranges to determine which sample to trigger.
HQ	**High Quality** Turns on high quality mode, that aliases less when resampling at high and low frequencies. If off the sampler uses less CPU heavy interpolation.
LOOP	**Loop** Selects global loop type. None, Forward, Backward and Pingpong. Loops between the start and start + length positions.
POS	**Position** Instantly changes the playing position when playing a sample.
SAMPLE	**Sample** Changes the currently played sample. When only one sample is loaded or Sampler is in Multi-mode, this knob has no effect.

Buttons

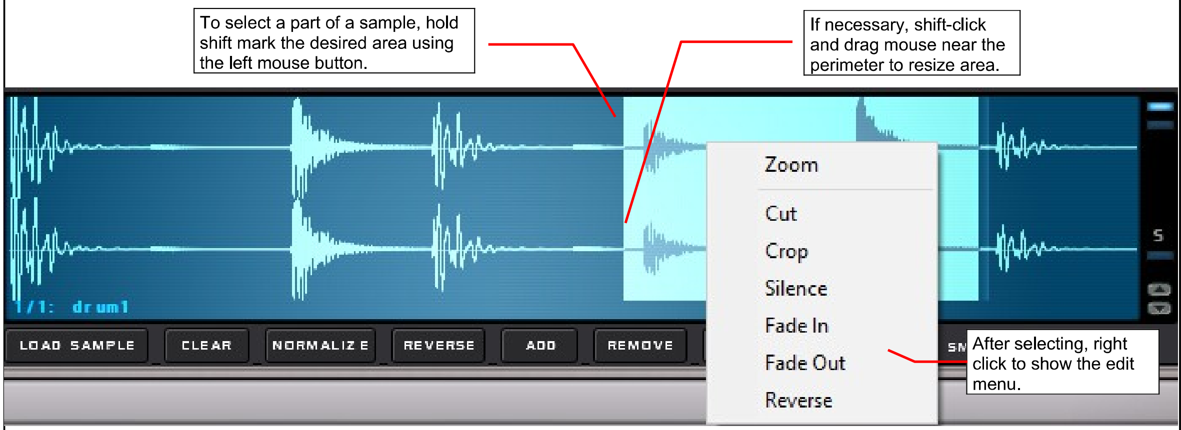
LOAD SAMPLE	Load (and replace) a .WAV file from disk after selecting it from the file browser. You can also select and drag files from anywhere and drop them on the Sampler module.
CLEAR	Clears the current sample slot, removing the sample from memory.
NORMALIZE	Maximizes the sample so the highest peak uses all the available dynamic range.
REVERSE	Turns the sample around so it plays backwards.
ADD	Adds a new sample slot.
REMOVE	Removes the currently selected sample slot.
RECORD	Clears the sample and records a new sample directly from the input stream. Click record again to stop recording.
IMPORT	Imports .SF2/.SFZ multi-samples, including sustain loops and key/velocity ranges.

Sample Display – Standard



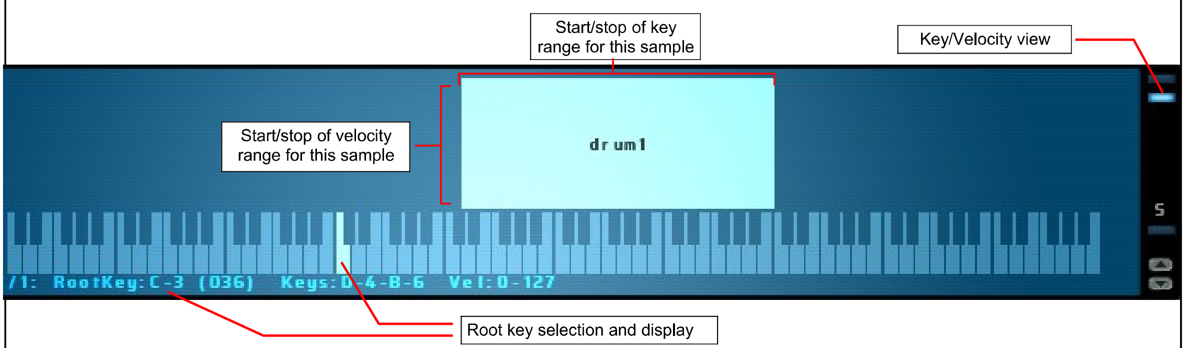
1.	**Play position marker** Shows the current play position when the sampler is playing.
2.	**Sustain start/stop markers** Specifies the position in the sample to start and stop sustained looping. The markers are only shown when sustain mode is enabled. Press left mouse and drag the handles of the markers to change their position.
3.	**Sample view** Click to show the sample (this view).
4.	**Sample index out of total number of loaded samples (index/total)** Displays the index of the shown sample out of the total number of loaded samples.
5.	**Sample name** Name of the shown sample.
6.	**Sustain on/off** Switch sustain mode on/off. In sustain mode the sample will loop continuously between the positions indicated by the sustain markers, as long as a key is held.
7.	**Sample select up/down** Use arrows to browse sample slots. If shift is held, the arrows will browse up/down in the disk folder where the sample resides.
	**Left press mouse and drag up/down** Zoom in or out in sample at the selected position.
	**Hold shift + left press mouse and drag** Selects a part of the shown sample (see next page).
	**Right Click on sample display** Lets you rename the currently shown sample, or C*rossfade Sustain Loop:* This will alter the sample to create a seamless sustain loop by crossfading the loop with the area leading up to the loop. The area after the loop is also crossfaded with the loop.
	**Mouse Wheel + Optional Drag** Zoom and pan horizontally.
	**Double Click Mouse Wheel** Reset Zoom.

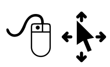
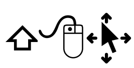
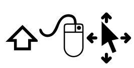
Sample Display – Sample Selection and Editing



ZOOM	**Zoom** Zooms to show only the selected part of the sample.
CUT	**Cut** Removes the selected part of the sample.
CROP	**Crop** Removes everything but the selected part of the sample.
SILENCE	**Silence** Replaces the selection with silence.
FADE IN	**Fade In** Fades the selection from zero to full volume.
FADE OUT	**Fade Out** Fades the selection from full volume to zero.
REVERSE	**Reverse** Reverses the selection so it plays backwards.

Sample Display – Root Key, Key Range and Velocity Range



	**Left click on the drawn keyboard** Selects new root key. The root key selects the pitch at which the sample is played "as is", i.e., without any stretching or resampling. All resampling is done according to the chosen root key.
	**Shift + Left click in key/velocity field and drag** Sets start of key range and maximum velocity. In multi mode a sample will only be activated when hitting keys inside the key/velocity range. Velocity ranges cannot overlap.
	**Shift + Right click in key/velocity field and drag** Sets end of key range and minimum velocity. In multi mode a sample will only be activated when hitting keys inside the key/velocity range. Velocity ranges cannot overlap.

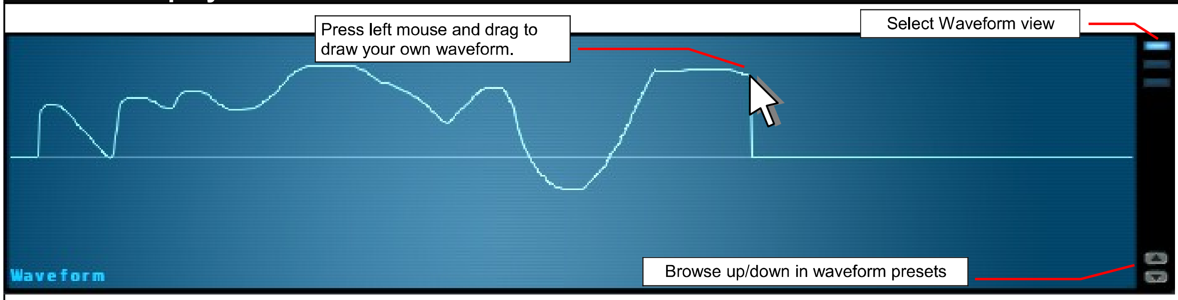
Additive

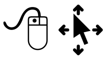
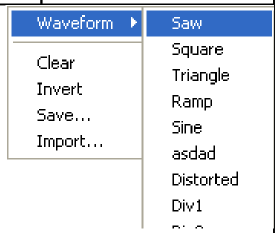
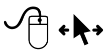
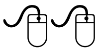
Generator

Additive is a general waveform oscillator. Unlike Osc1 and Osc2 which have a fixed set of waveforms, any waveform is created by directly editing the waveforms harmonics. A waveform can simply be drawn using the mouse or edited by changing the magnitude and phase of up to a total of 1024 harmonics (Use the switch buttons to the right of the waveform display to change between waveform, magnitude and phase editing). Waveforms can also be saved as presets and imported from external .wav files.

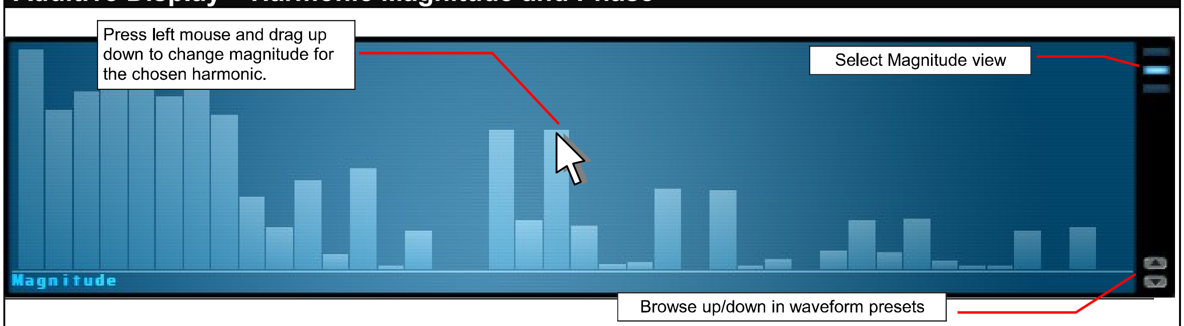
Parameters	
START	**Phase Start** Sets the start position in the waveform (Phase init).
PHASE	**Phase** Sets the phase value, or where we are in the cycle. Modulate this value to do phase modulation.
DETUNE	**Detune** Adjust the frequency up or down steplessly one semitone.
FREQ	**Frequency** Adjusts the frequency up or down continuously one or more octaves. Octave range and optional snap can be set through the right-click menu.
TRIGGER	**Trigger** When enabled the signal will phase restart on note-on events.
MONICS	**Harmonics** Sets the total number of harmonics used for waveform construction. The higher the value the more high frequency contents the waveform will contain. At max setting (1024) the waveform is allowed to generate even higher harmonics.
SCALE	**Scale** Scales the magnitude of the harmonics.
OFFSET	**Phase Offset** A Phase *Offset* adjustment knob appears whenever the phase editor is active. Turn the knob to adjust the sliding phase offset for all harmonics, so low harmonics is given a short offset while high harmonics is given a high offset. This affects the waveform significantly, as it slides the phase relation and alters transients, without changing the individual harmonics.
GIBBS	**Gibbs** Controls the amount of *Gibbs*-effect to allow in the waveform construction. The Gibbs phenomenon is usually an unwanted side-effect of a clamped Fourier series. It appears as if the sines near discontinuous overshoot, and is therefore also referred to as *ringing*. The overshoot is a consequence of trying to approximate a discontinuous function with a finite sum of continuous sines. Musically, however, it can give some interesting results, especially combined with

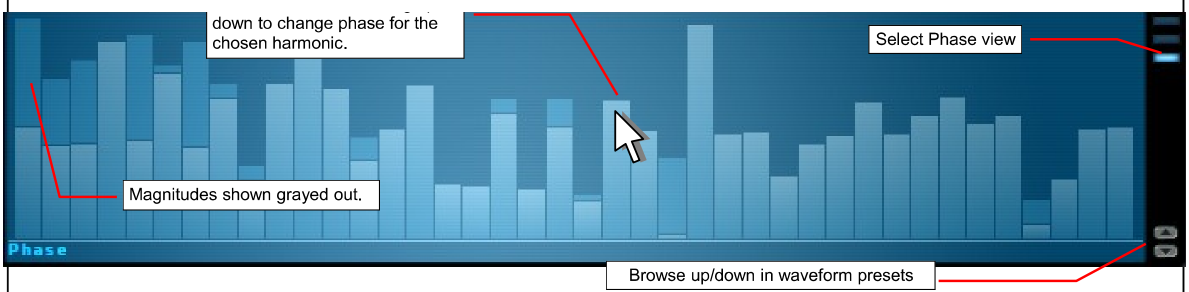
Additive Display – Waveform

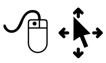
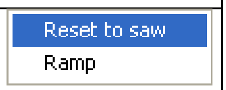


	**Left click + Drag** Draw your own waveform. When the mouse button is released the drawn curve is converted into the chosen number of harmonic magnitudes and phases.
	**Right Click** Shows the right click menu:  WAVEFORM: Directly select a built-in/user waveform preset. CLEAR: Resets waveform to silence. INVERT: Flips waveform upside down. SAVE...: Save drawn waveform to disk as a user-preset. IMPORT...: Imports a .WAV file and converts it into a waveform. Alternatively drag&drop a file onto the module.
	**Mouse Wheel + Optional Drag** Zoom and pan horizontally.
	**Double Click Mouse Wheel** Reset Zoom.

Additive Display – Harmonic Magnitude and Phase





	**Left click + Drag** Change the magnitude or phase of one or more harmonics.
	**Right Click** Shows the right click menu:  RESET TO SAW: Set magnitude or phases to saw profile. RAMP: Fades magnitudes or phases to zero.

Pad

Generator

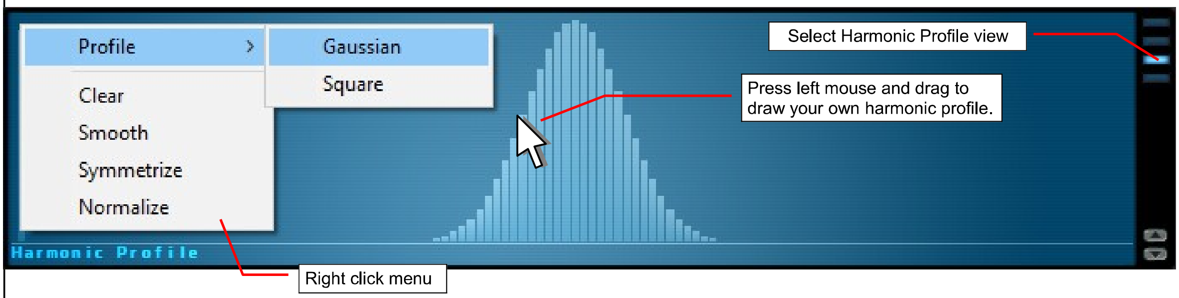
Pad is a stereo generator suitable for slow evolving pad type sounds. Like the Additive module, a core waveform is constructed using harmonics (please see the Additive module for waveform and magnitude editing details). However, unlike the Additive module each harmonic controls a whole range of partials, using a so called *harmonic profile*. The harmonic profile is an editable curve responsible for smearing out the generated frequencies. The produced partials are given random phases, causing the sound to "drift". The result is a wide, slowly evolving, pleasant sounding waveform. The size of the harmonic profile can be changed using the bandwith and bandwith scale parameters. A frequency response curve can be changed to adjust the overall frequencies of the produced waveform.

The generated sound is precalculated and stored in memory as a wavetable. It is possible to adjust the size of this wavetable, thereby controlling the sound quality. Higher quality yields better sound, but also takes longer to precalculate, resulting in higher editing latency[*].

Parameters	
START	**Phase Start** Sets the start offset into the generated waveform. Set on note trigger.
STEREO	**Stereo** Control the offset used for the right channel, to create a stereo effect when out of phase. A setting of 0 means no offset, and hence no stereo.
DETUNE	**Detune** Adjust the frequency up or down steplessly one semitone.
FREQ	**Frequency** Adjusts the frequency up or down continuously one or more octaves. Octave range and optional snap can be set through the right-click menu.
TRIGGER	**Trigger** When enabled the signal will phase restart on note-on events.
MONICS	**Harmonics** Sets the total number of harmonics used for waveform construction. The higher the value the more high frequency contents the waveform will contain.
BW	**Bandwith** Changes the size of the harmonic profile in the frequency spectrum. A higher setting means a wider profile, yielding a wider sound.
BWSCALE	**Bandwith Scale** Controls how the bandwidth changes with frequency. A high setting gives a wider, increasing profile, yielding more spread in the higher frequencies and producing a softer sound. A low setting gives a slimmer, reducing profile for higher frequencies, producing a harsher sound.

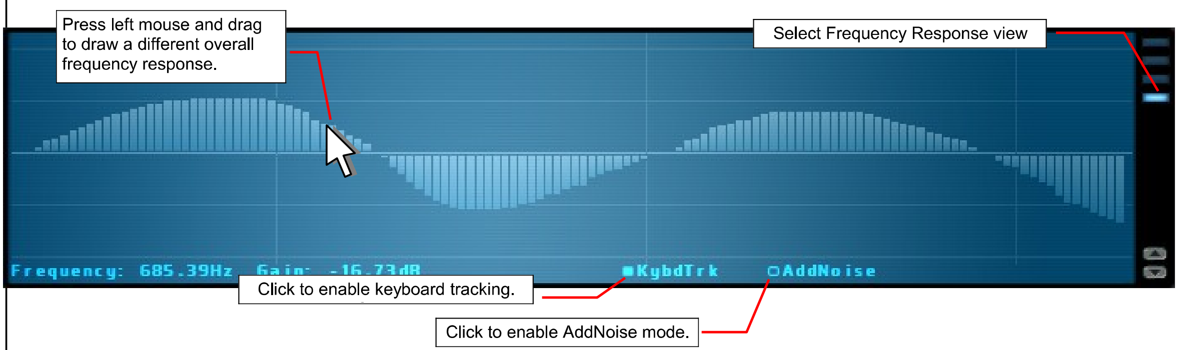
[*] Internally the Synth uses a separate thread to compute the resulting wavetable (On multi-core systems this even runs on a separate CPU), so you will not feel any latency in the GUI. Sound-wise you might however experience a slight delay before the changes you make are actually heard.

Pad Display – Harmonic Profile



	Left click + Drag Draw your own harmonic profile. When the mouse button is released the profile is applied.
	Right Click Shows the right click menu: PROFILE: Select one of the built-in harmonic profile presets. CLEAR: Resets profile to silence. SMOOTH: Lowpass filters the profile a bit to make the curve extra smooth. SYMMETRIZE: Force profile to be symmetric by mirroring its left side. NORMALIZE: Maximizes the profile so the highest peak uses all the available space.
	Mouse Wheel + Optional Drag Zoom and pan horizontally.
	Double Click Mouse Wheel Reset Zoom.

Pad Display – Frequency Response



	Left click + Drag Draw a different overall frequency response curve.
	Right Click + Drag Reset one or more frequency bands to their default (0).
	Mouse Wheel + Optional Drag Zoom and pan horizontally.
	Double Click Mouse Wheel Reset Zoom.
KYBDTRK	**Keyboard Tracking** When enabled the frequency response curve is applied in such a way that the lowest editable frequency corresponds to the lowest played frequency. When disabled the curve is applied regardless of the played key.
ADDNOISE	**Add Noise** When enabled the frequency response curve is used to add noise to the generated waveform instead of modifying its frequency content.

Granular

Generator
NEW IN V.2

The Granular module offers built-in *granular synthesis*. Granular synthesis is a unique form for sample-based synthesis that takes a sample and chops it up into small clips, called *grains*. These grains can are then played back at a defined rate, but can be altered individually with respect to frequency, and shape, and - most importantly – independently from the samples playback speed. This means that it is possible to slow down the sample, without changing its pitch, as well as changing its pitch without changing the playback speed. A whole range of interesting possibilities arise when each grains frequency, position, size and panning parameters are also modulated, resulting in a completely different sounds than the source sample.

Parameters	
SIZE	**Grain Size** Sets the size of each individual grain.
RATE	**Grain Rate** Controls the grain spawn rate.
SHAPE	**Grain Shape** Controls the shape of each grain, from rectangular (0) to triangular (127). For overlapping grains, triangular will give most seamless result.
SPEED	**Speed** Controls the playback speed of the sample in Auto mode
TRIGGER	**Trigger** When enabled the sample will restart on note-on events. When off the sample will play continuously (free running).
MULTI	**Multi** Turns on multi sampling mode. In this mode the module uses the assigned key/velocity ranges to determine which sample to play from.
SYNC	**Tempo Sync** Syncs the rate frequency to the host tempo.
HQ	**High Quality** Turns on high quality antialiasing.
SAMPLE	**Sample Index** Choose playback sample index in non-multi mode.
MODE	**Playback Mode** [Auto/Manual] In **Auto** mode the sample position is advanced automatically, based on the speed parameter. In **Manual** mode the playback position is completely controlled by the position knob, and the speed parameter has no effect.
POS	**Grain Position** Controls the grain playback position within the sample.
FREQ	**Grain Frequency** Controls the grain playback frequency.
PAN	**Pan** Controls stereo panning of each grain in stereo mode.
STEREO	**Stereo** Controls stereo spread by offsetting the left and right channels in stereo mode.
DIFFUSION	**Diffusion** Enables diffusion (low amplitude random modulation) for **Pos**, **Freq**, **Size**, and **Pan**.

Noise

Generator

The Noise module generates Pink, White or Brown noise. Noise is especially useful for synthesis of ocean waves and wind, for percussion or simply to add some extra "bite" or analogueness to leads and basses. By definition noise is (digitally speaking) a series of random numbers containing almost all frequencies. The different flavors determine how the noise rolls-off over the frequency spectrum. White noise is the purest (and harshest) type of noise without any filtering. Pink noise rolls off linearly towards the high end at 3dB/Octave. Brown noise rolls-off at 6dB/Octave. One can optionally enable cycle mode to force the generated noise to repeat at an interval, thus forming a waveform similar to an oscillator, but randomized.

Parameters	
AMP	**Amplifier** Sets the overall volume.
SEED	**Seed** To control the randomness, this knob can be used to set the initial seed sent to the (pseudo) random algorithm.
LOWPASS	**Lowpass** Controls one pole (12dB/Octave) lowpass filtering of the noise. When set to 127, all frequencies pass.
HIGHPASS	**Highpass** Controls one pole (12dB/Octave) highpass filtering of the noise. When set to 0, all frequencies pass.
TRIGGER	**Trigger** When enabled the signal will phase restart on note-on events.
CYCLE	**Cycle** Enables cycle mode, where the generated waveform is reset at an interval corresponding to pitch set by the controller.
NOISE TYPE	**Noise Type** Choose between: **White:** Clean noise without filtering (even distribution of energy). **Pink:** Filtered white noise with 3dB roll off per octave. **Brown:** Filtered white noise with 6dB roll off per octave.

Input

Generator

The Input module feeds the sound signal sent from the host into the Synth. This is normally used when the synth is used as an insert effect in the host application. The module receives audio input sent from the host in two separate streams. This can be used, e.g., for doing sidechain compression: One stream for the signal to be compressed and another stream for the sidechain. The module always receives a stereo signal, but you can force it to only output the left or right mono channel, e.g., if you want to do the stereo mixing yourself.

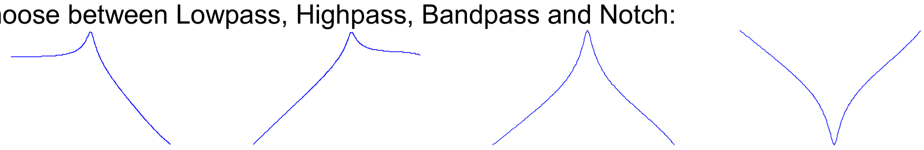
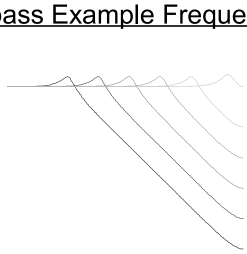
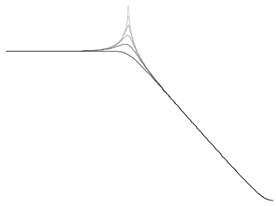
Parameters	
VOLUME	**Volume** Adjusts the volume of the input signal. Centered means 0dB (no adjustment). Turning the knob all the way up boosts by +6dB. Turning it all the way down means silence (-∞dB).
PAN	**Pan** When in stereo mode this knob pans the signal left or right.
INPUT	**Input Stream** Selects input stream.
MODE	**Mode** Selects which part of the input stream that should be fed in: 2 channel **Stereo** (default), **Mono Left**, **Mono Right** or **Mono Left+Right**. If a mono mode is selected, but the Input module is in stereo, the same mono signal is duplicated to both channels.

SVF/SVF2/SKF/Zolzer/Moog/Moog2/Acid/MS20

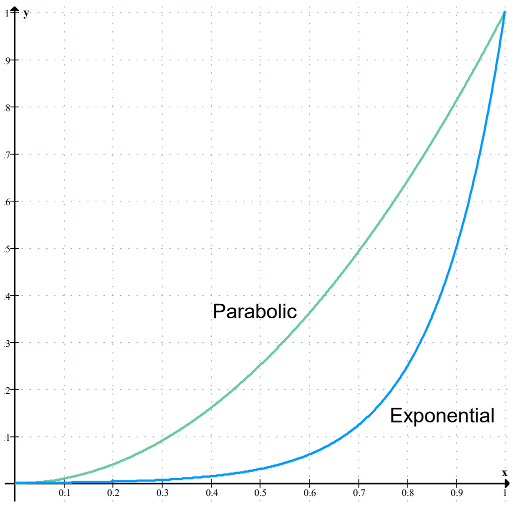
Filter

The SVF and SVF2 (State Variable Filter), SKF (Sallen-Key Filter), Zolzer, Moog, Moog2, Acid and MS20 modules are simulated analog resonant multi-mode filters (i.e., Voltage Controlled Filters, VCFs). They share the same parameters: Cutoff, Resonance, Mod and Kybd (Keyboard Tracking), Saturation and Drive, and have a selection of filter types: Lowpass, Highpass, Bandpass and Notch (Bandreject). The filters have subtle but different sound characteristics: The SVF and SKF are the "standard" synth filters you can turn to whenever general filtering is needed. Zolzer can sometimes sound more round and analog, while Moog has a *Moog*'ish sound to it, meaning that it has a steep cutoff and is remarkably self-resonant. Finally, Acid emulates the sound characteristics of Roland's famous TB303 bass unit, and MS20 the analog sound of the Korg MS20 Filter. Because the filter modules share the same parameters, parameter settings are retained when switching between them.

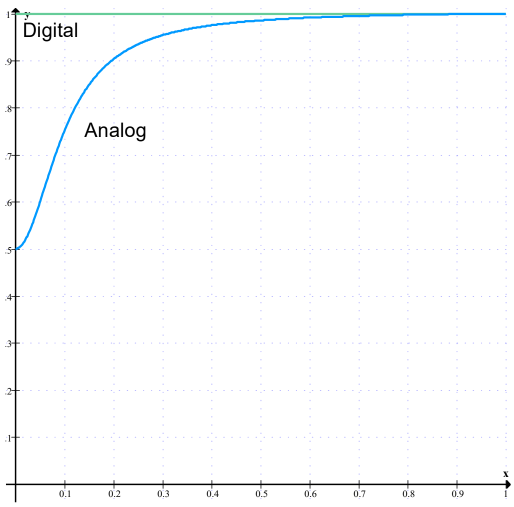
Parameters	
CUTOFF	**Cutoff Frequency** Selects the cutoff point, which is the frequency in Hertz where the filter will start to attenuate. The right click menu contains an Expression option and a Cutoff Tuning setting (see below).
RESO	**Resonance** Selects the amount of boost to apply at the cutoff (at around –3dB). The right click menu contains an Expression option and a Resonance Tuning setting (see below).
MOD	**Modulation Amount** Adjusts the amount of modulation to apply to cutoff. If cutoff isn't modulated this knob has no effect. Optional modulation factors are 1(default), 2, 4 and 8.
KYBD	**Keyboard Tracking** Sets the amount of keyboard tracking from –200% to 200%. This offsets the cutoff frequency with respect to the incoming frequency control signal. Setting this to 100%, means that the outgoing sound will have the same timbre regardless of the played pitch. Without a frequency control signal, the **Kybd** knob will have no effect. Hence, a filter should be placed <u>before</u> any controllers in the signal chain in order to perform keyboard tracking. The **Base Key** (default:C1) from which the Keyboard Tracking is determined can be setup through the right click menu.
SATURATION	**Saturation** (Knob appears when the *SAT* LED is selected) Controls the amount of saturation of the internal feedback inside the filter. Saturation sets the mix ratio between the unsaturated and the saturated signal. A parameter setting of 0 means no saturation (default).
DRIVE	**Drive** (Knob appears when the *SAT* LED is selected) Controls the drive of the internal saturation of the filter. Drive sets the prescaling of the unsaturated signal feeding into the internal soft saturator inside the filter, such that a higher value means more distortion. For high values, the distortion can cause the filter to overdrive and self-resonate.

Parameters (Continued)	
2/4 POLE 4/8 POLE	**Filter Steepness** The number of poles determine the steepness of the filter, A 2 pole filter attenuates 12dB/Octave, 4 pole: 24dB/Octave and 8 pole: 48dB/Octave.
TYPE	**Filter Type** Choose between Lowpass, Highpass, Bandpass and Notch:  Lowpass Highpass Bandpass Notch <u>SVF Lowpass Example Frequency Responses:</u>  Lowpass Cutoff Sweep  Lowpass Resonance Sweep

Cutoff & Resonance Tuning Curves



Cutoff Tuning



Resonance Tuning

CUTOFF TUNING	All Cutoff parameters in the synth have a Tuning option, that maps the parameters linear setting to a frequency through a Tuning curve. The **Exponential** curve (default) maps the frequency setting *exponentially* similar to how humans naturally perceive pitch. This gives equal amount of control in the high-end of the spectrum as in the low-end, but it is never able to reach 0Hz. The **Parabolic** curve is a simpler legacy option, that gives less low-end control, but if needed does reach 0Hz.
RESONANCE TUNING	The Resonance parameter has a Resonance Tuning setting that optionally tones down the Resonance at low-frequencies, as defined by the **Analog** Resonance tuning curve. This mimics the response of old analog filters. Default is a flat **Digital** resonance response.

EQ3

Filter

EQ3 is a basic 3 band equalizer with adjustable crossover frequencies, and 2 and 4 Pole modes. Like any equalizer it can be used to to boost or attenuate certain frequencies in order to shape a sounds timbre.

Parameters	
LOW	**Low Gain** Boost or cut the Low-frequency range. Turning the knob all the way up boosts by +6dB. Turning it all the way down removes that range completely (-∞dB).
MID	**Mid Gain** Boost or cut the Mid-frequency range. Turning the knob all the way up boosts by +6dB. Turning it all the way down removes that range completely (-∞dB).
HI	**Hi Gain** Boost or cut the High-frequency range. Turning the knob all the way up boosts by +6dB. Turning it all the way down removes that range completely (-∞dB).
AMOUNT	**Amount** Controls the mixed amount of EQ. All the way up means 100% (Full EQ), while center means 50% EQ and 0 means no EQ (Bypass).
2/4 POLE	**2/4 Pole** Switch between 2 Pole (12dB per octave) and 4 Pole (24dB per octave) filters.
FREQ1	**Freq1** (Knob appears when the *FREQS* LED is selected) Controls the crossover frequency (100Hz to 2kHz) between Low and Mid Bands.
FREQ2	**Freq2** (Knob appears when the *FREQS* LED is selected) Controls the crossover frequency (2kHz to 16kHz) between Mid and Hi Bands.

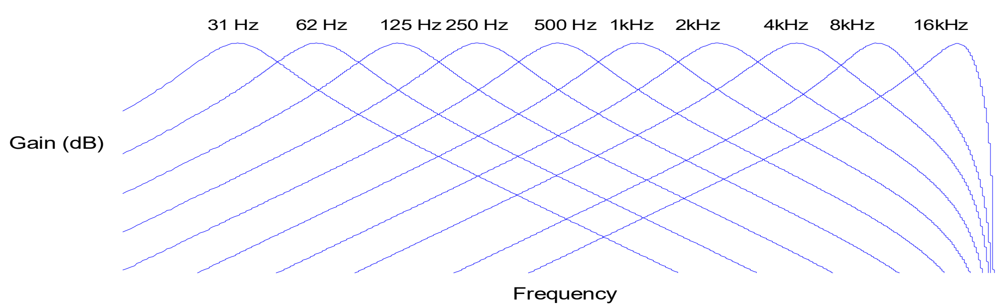
EQ10

Filter

EQ10 is a basic 10 Band digital graphic equalizer (a parallel bandpass filter bank) that can be used to boost or attenuate ("cut") certain frequencies in order to shape a sounds timbre. It has 10 parameters, one slider for each band, starting from the left: 31Hz (low frequency) to the right: 16kHz (high frequency). The frequency response of the individual bands (shown below) has a characteristic dive near the Nyquist frequency, which is typical digital EQ behavior. To get analog behavior use EQ31 instead.

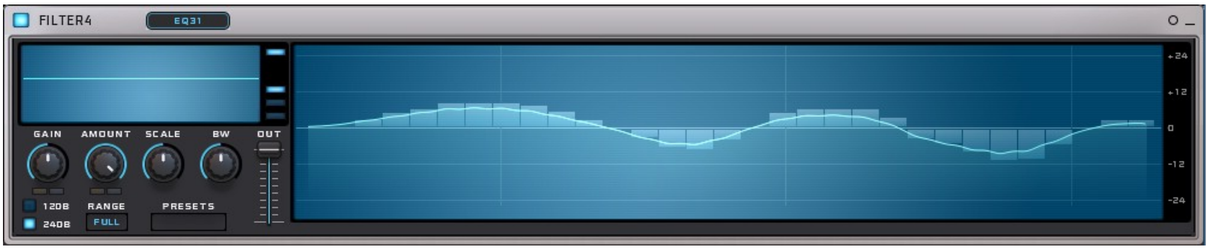
Parameters	
1-10	**Frequency boost/cut** Sliders that adjust the boost or cut for a frequency range around the center frequencies: 31Hz, 62Hz, 125Hz, 250Hz, 500Hz, 1kHz, 2kHz, 4kHz, 8kHz, and 16kHz. Turning a slider all the way up boosts by +12dB. Turning it all the way down removes that range completely (-∞dB).

EQ10 Frequency Plot



EQ31

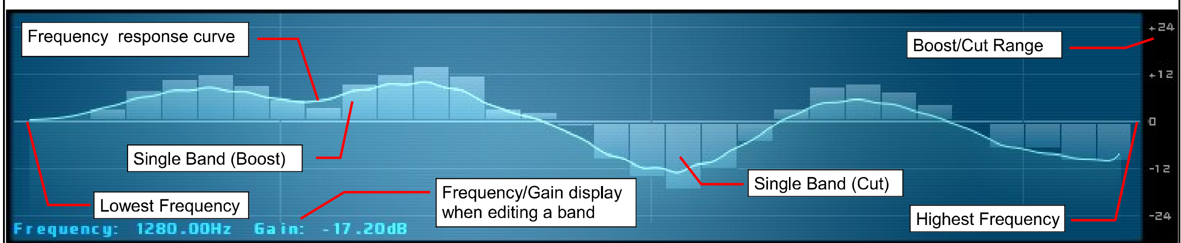
Filter

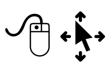
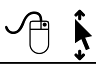


EQ31 is a 31 band simulated analog graphic equalizer (serial peaking EQ filter bank) that like EQ10 can be used to adjust the timbre of a sound across the whole frequency range. The main difference to EQ10 is that EQ31 has a higher density of bands and an analog response. Moreover, each band can have unique boost/cut as well as bandwith settings, and their combined frequency response is shown graphically. However, unlike EQ10, the individual band parameters cannot be modulated.

Parameters	
GAIN	**Gain** Increases or reduces the overall volume of the output. This is sometimes necessary when cutting or boosting a lot.
AMOUNT	**Amount** Sets how much of the EQ'ed signal you wish to hear by adjusting the mix ratio between the unfiltered and the filtered signal.
SCALE	**Scale** Scales all bands by a common scale-factor to either increase or decrease their effect. Default is centered (scale=1).
BW	**Bandwith** Scales all bands bandwith by a common scale-factor. The higher the bandwith the wider the range of frequencies that each band affects. Turning the bandwith up reduces gaps between the bands and gives a smoother overall frequency response. Turning it down yields a more rippling frequency response. Default is centered for standard EQ behavior.
12/24 DB	**12/24 dB** Set Boost/Cut amount in dB. Choose between 12dB (gentle) and 24dB (hard) EQ.
RANGE	**Frequency Range** A dropdown selection of Low (20Hz-500Hz), Mid (500Hz-5kHz), High (5kHz-22kHz) and Full frequency ranges (20Hz-22kHz), where the band frequencies start/stop.
PRESETS	**Presets** A dropdown selection of useful predefined band settings.

EQ Display



	**Left Click or Click+Drag** Adjust the individual band positions, similar to adjusting the sliders on a standard EQ.
	**Right Click + Drag up/down** Adjusts bandwith for individual bands.

Formant

Filter

The Formant module is a special kind of filter that mimics the human voice by simulating resonant peaks in the frequency spectrum, or so called "formants". Formants appear in human speech and singing due to the resonant frequencies of the vocal tract but also in some musical instruments. In human speech they are essential to distinguish between vowels, since each vowel has a unique frequency pattern. Therefore the module has two knobs for choosing a vowel pattern and a blend knob to blend between these patterns.

Parameters	
VOWEL 1/2	**Vowel 1/Vowel 2** Choose a vowel (A,E,I,O,U). The vowels blend smoothly into each other.
BLEND	**Blend** Choose the mix ratio between the two vowel patterns. 0 means 100% vowel 1. 127 means 100% vowel 2.
AMOUNT	**Amount** Sets the dry/wet amount of filtered output to mix with the input.

Shelving

Filter

Shelving is a simple filter module containing two 2 pole shelving filters: Lowpass (High shelf) and Highpass (Low shelf) for cutting the high end and the low end of a signal respectively.

Parameters	
LP	**Lowpass Frequency** Sets the lowpass frequency, where the high cut should start.
LP GAIN	**Lowpass Gain** Sets how much of the lowpass signal to blend with the input.
HP	**Highpass Frequency** Sets the highpass frequency, where the low cut should start.
HP GAIN	**Highpass Gain** Sets how much of the highpass signal to blend with the input.

Comb, Allpass

Filter

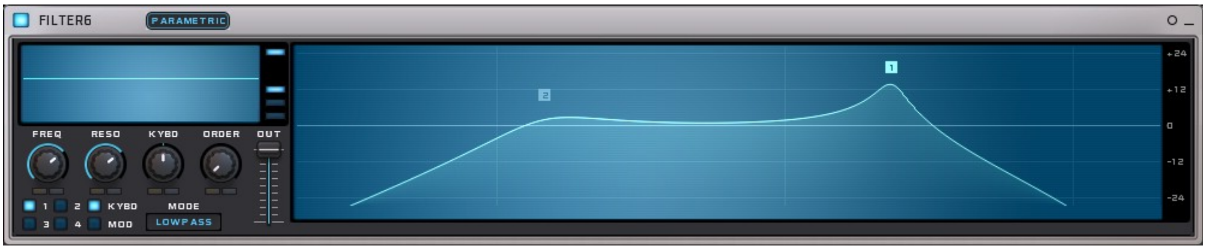
The Comb and Allpass filter modules share the same set of parameters. The Comb filter adds a delayed version of a signal to itself, causing phase cancellations. The result is a frequency spectrum with several regularly-spaced spikes that sort of look like a comb. The Allpass filter has similar structure, but contains an additional feed-forward element. This yields a flat frequency response, meaning that all frequencies pass but that their phases are smeared.

Both Comb and Allpass filters are traditionally used to construct reverberators. Still they have characteristic sound properties that make them interesting for synthesis. E.g., the comb filter can be interpreted as a simplified model of a physical resonator, and be used to simulate several non-persistently-excited decaying resonant instruments.

Parameters	
FREQ	**Frequency** Controls the time delay for both Comb and Allpass, since the delay is related to the frequency of the filters.
(SCALE)	**Frequency Scale** (only in Keyboard Tracking mode) Scales the incoming frequency control signal by a scale factor (0-2). Centered means no scaling.
FEEDBACK	**Feedback** Sets the amount of feedback, or how much of the delayed signal to send back into the filter.
DAMP	**Damp** Both Comb and Allpass feature an internal 1-pole lowpass filter for damping the feedback. Turned full left means no damping.
POLARITY	**Polarity** Controls the sign and scale of the feedback. Full right means feedback of +1, while full left equals –1.
KYBDTRK	**Keyboard Tracking** Forces the module frequency to match the incoming frequency control signal. The *Freq* knob is replaced with the Frequency *Scale* knob in this mode.

Parametric

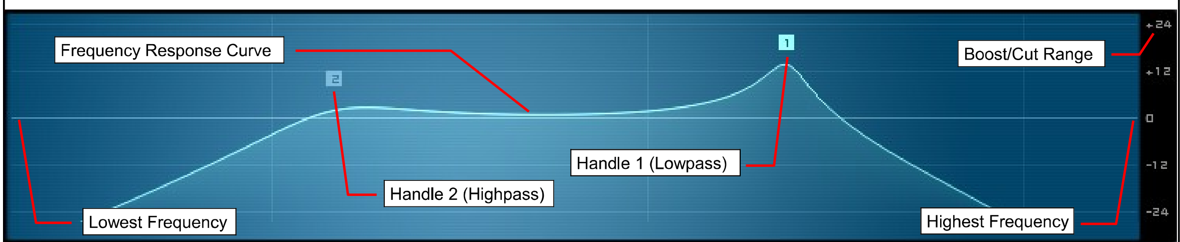
Filter

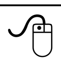
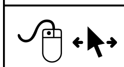
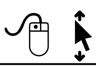
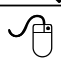


The Parametric filter is the most general and user-controllable filter of all the filters in the Synth. It features 1 to 4 simultaneous filters, placed in series, that each can be set to any of the 6 filter types. The four filters have tweakable Frequency-, Resonance-, Gain-, Order- and Modulation-settings. The module has a display for editing the filter settings (using a small square *handle* for each filter) as well as showing the resulting overall frequency response of the applied filters.

Parameters	
FREQ	**Frequency** Sets the cutoff frequency.
RESO	**Amount** Adjusts the boost to apply near the cutoff frequency.
GAIN/KYBD	**Gain / Kybd** For Peak and Shelving filters this knob sets the overall gain. For all other filters (lowpass, highpass, etc.) this knob controls keyboard tracking.
MOD	**Modulation Amount** Adjusts the amount of modulation to apply to cutoff. If cutoff isn't modulated this knob has no effect.
ORDER	**Order** Order changes the steepness of the filter steplessly from 2 pole to 8 pole.
1 / 2 / 3 / 4	**Filter 1, 2, 3, 4** These LEDs select the active filter.
GAIN / MOD	**Gain/Kybd / Mod selection** Choose between Gain/Kybd or Mod knob access. The knob changes accordingly.
MODE	**Filter Mode** Set filter mode for the currently selected filter. Choose between: Off, Lowpass, Highpass, Bandpass, Notch, Peak, Lowshelf or Highshelf

Parametric Filter Display



	**Left Click on Handle** Selects that filter. The parameters are updated to reflect the change.
	**Left Click on Handle + Drag left/right** Changes Cutoff frequency for selected filter.
	**Left Click on Handle + Drag up/down** Adjusts resonance boost or gain for selected filter.
	**Right Click on Handle** Turns off that particular filter.

Amplifier

Similar to a VCA on an analog synth the Amplifier module is a simple device that changes the amplitude/volume/loudness of a signal. It features a velocity control and a Gate mode for turning the signal on and off on MIDI events. Furthermore, since channel volume is related to panning, the module also features simple stereo panning.

An important property of the Amplifier is that it will skip receiving streams from modules if the amplification is 0 for more than half a second. This can reduce CPU load considerably, especially in polyphonic patches. Furthermore, this also ensures that patches will consume 0% CPU when not playing (and when the GUI is closed).

Parameters	
AMP	**Amplification** Changes the volume of the incoming signal. Centered means no change in volume. A higher setting boosts, while a lower setting reduces the volume. This is a useful knob to modulate using, e.g., an ADSR envelope or a LFO in order to change the volume of a patch over time. The right-click-menu offers a **Declick Ramp** option that controls the amplification declick ramp time (See *Declick*).
VELOCITY	**Velocity** MIDI velocity tells something about how hard you press a key on your MIDI keyboard, and is usually used to control the resulting volume of a patch. This knob sets how much velocity should affect the volume setting. 0 means no effect, while 127 means full velocity control.
PAN	**Pan** When in stereo mode this knob pans the signal left or right.
MOD	**Amp Modulation Amount** Controls how much modulation to apply to the Amp knob. If Amp isn't modulated this knob has no effect.
DECLICK	**Declick** When ON the module ramps quick volume changes to avoid clicks in the sound. Ramping time is configurable. See *Amp* knob.
PANORAMA	**Panorama** When ON the module samples and freezes the current pan position on note-on events.
TRIGGER	**Trigger** When ON the module responds to MIDI events, taking into account note-on, note-off, and velocity settings. When off the module just works as a standard amplifier.
GATE	**Gate** When ON the module only lets the incoming signal pass when between note-on and note-off events. Otherwise it outputs silence. Useful for when not modulating the Amp knob. Default is ON.

Mixer

Amplifier

Mixer is a special module for mixing two (and only two) incoming signals. By default the synth automatically mixes all signals sent to a module, but in this case you can control the type of mixing to use.

Parameters	
MIX	**Mix** Changes the mix ratio between the incoming signals. Centered equals standard 50/50 mixing ratio.
AMP	**Amplification** Changes the volume of the incoming signal. Centered means no change in volume. A higher setting boosts, while a lower setting reduces the volume.
PAN	**Pan** When in stereo mode this knob pans the signal left or right.
MOD	**Mix Modulation Amount** Controls how much modulation to apply to the Mix knob. If Mix isn't modulated this knob has no effect.
TYPE	**Mix Type** Chooses the Mixing type to use to combine the two signals: **Add**: Linearly Add the two signals. This standard mixing is the default. **Sub**: Subtract the two signals. **Ringmod**: Multiply the two signals. **2xMono**: Take two mono signals and put them in the left and right channel. **Xor**: Perform a per-sample binary Xor operation on the two signals. **Greater**: Perform per-sample comparison the two signals, and choose the largest.

Stereo

Amplifier

Stereo is a simple Amplifier module for changing the volume of the left and right stereo channels individually. Moreover, it can pan (adjust balance) of the incoming sound in the stereo image. Panning can be done with respect to 4 different panning laws.

Parameters	
LEFT	**Left (0 to 200%)** Controls the volume of the left stereo channel.
RIGHT	**Velocity (0 to 200%)** Controls the volume of the right stereo channel.
PAN	**Pan** When in stereo mode this knob pans the signal left or right. The volume of the left and right channel is changed with respect to the chosen pan law.
MOD	**Left/Right Modulation Amount** Controls how much modulation to apply to the Left & Right knobs. If Left or Right aren't modulated this knob has no effect.
STEREO INVERT	**Stereo Invert** Flips left and right channel, so left becomes right and vice versa.
ANTIPHASE	**Anti Phase** When ON flips the direction of the right knob, so the left and right channel run out of phase. Useful for creating interesting stereo effects with a single LFO.
PAN LAW	**Pan Law** Panning laws (originally from analog mixers) internally determines how much of each source signal is sent to the two stereo channels. They compensate for the fact that the sum of the two channels can sound louder when centered than when panned full left or full right. Instead a logarithmic gain change drop of: –0dB, -3dB, -4.5dB or –6dB is typically used at the center.

Inverter

Amplifier
NEW IN V.2

Inverter is a simple Amplitude Inverter module. It allows for both Unipolar and Bipolar amplitude changes, making it useful for Amplitude Modulation. It can also be used inverting a modulation signal or for adding a DC offset to a signal.

Parameters	
AMP	**Amp** (dB) Controls the (unipolar) amplitude boost. Centered means no changes. Full right means +6dB, and full left means -∞dB.
POLARITY	**Polarity** Controls the (bipolar) amplitude polarity. Full right means positive (+1) and full left means negative (-1) polarity (inverted signal). You can modulate this for bipolar AM.
MOD	**Mod** Controls the Modulation scale of the Polarity knob.
OFFSET	**DC Offset** Adds a DC offset to the final output, either positive or negative. 0 means no DC.

Mid/Side

Amplifier
NEW IN V.2

Mid/Side is a simple stereo splitter. It splits the signal into a mono-only part called *Mid*, and a stereo-only part called *Side*, and allows you to control the volume of these signals separately. In addition it allows you to change the polarity of the side signal, as well as control the overall panning of the output. The module is useful for doing creative stereo processing. For instance, by using the module it is possible to isolate the stereo only part of an input signal, then process this signal separately until finally mixing it with the mono signal.

Parameters	
MID	**Mid** (dB) Controls the volume of the mid signal. *Mid* is the mono-only part of the signal.
SIDE	**Side** (dB) Controls the volume of the side signal. *Side* is the stereo-only part of the signal.
POLARITY	**(Side) Polarity** Changes the polarity of the side signal.
PAN	**Pan** When in stereo mode this knob pans the signal left or right.
STEREO / MONO	**Stereo / Mono** Switches between **Stereo** mode (default) or **Mono** mode which converts the mixed Mid/Side result to Mono before output.
MODE	**Mode** Choose between **Mix** (default) which mixes the Mid and Side signals. **Mid Only** which soloes the Mid signal and removes the Side signal completely or **Side Only** which soloes the Side signal and removes the Mid signal completely.

NotePitch

Controller

The NotePitch controller is an essential and useful module for controlling generator pitch, note triggering and polyphony. Moreover it has simple transpose, finetune, and portamento (glide) functionality. A controller's job is to construct a frequency signal that is sent to all modules feeding their output to that controller. The knobs on the NotePitch module affect this internal control signal.

Parameters	
NOTE	**Note** Transposes the internal signal whole semitones up or down. Base-note is shown in the display's text area. The Right click menu offers a **Tuning** option to detune the root key of the scale (default is A=440Hz), and a **Note-Min/Max** option that allows for setting the range of notes that the module responds to. Default is full MIDI range (C-0 to G-10).
OCTAVE	**Octave** Transposes the internal signal whole octaves up or down. Base-note is shown in the display's text area.
TUNE	**Tune** Finetunes the internal signal steplessly one or more semitone up or down. Choose between 2, 12 or 24 semitone range by right clicking on the knob. The right click menu also offers an *Enable Pitchbend* option (on by default). When on the module will handle MIDI pitchbend events automatically using the chosen semitone range. Finally, the right click menu offers an *MPE PB Range*: The MIDI Polyphonic Expression Pitchbend Range controls the amount of Pitchbend for MPE controlled notes when MPE is enabled by the host (±48 semitones is default).
PORTA	**Portamento/Glide** Changes how fast the internal frequency signal changes. Turned full left means no portamento (or glide). The right click menu offers range scales and two portamento modes: *Fixed* or *Auto*: In Fixed mode, glide always takes place. In Auto mode (default), only overlapping notes will glide.
TRIGGER	**Trigger** When ON the module listens and reacts to note-on MIDI events. It changes the frequency of the internal signal based on the currently played MIDI note.
MODE	**Mode** Chooses Polyphony mode. Choose between: **Mono**: 1 voice only. **Poly** 2-16: Polyphonic with 2 to 16 voices. **Legato**: 1 voice, without re-trigger as long as at least one key is held.

Frequency

Controller

The Frequency module lets you control the internal frequency control signal by using a single knob. This is great for fast frequency sweeps or for transposing to odd frequencies. It also does note triggering and portamento, but unlike NotePitch it doesn't support polyphony. However, you can easily work around this by routing the output from the Frequency controller to a NotePitch controller.

Parameters	
FREQ	**Frequency** Sets the frequency continuously (or snapped to semitones when in snap mode) to a frequency in the range 0.2Hz to 16kHz (about 8 octaves).
FINE	**Finetune** Finetunes the frequency up or down 40 cents.
SCALE	**Tune** Scales the incoming note frequency by a common scale factor. Centered means no scale. 0 multiplies by zero, while 127 multiplies by 2.
PORTA	**Portamento** Changes how fast the internal frequency signal changes. The right click menu offers two portamento modes: *Fixed* or *Auto*: In Fixed mode , glide always takes place. In Auto mode (default), only overlapping notes will glide.
TRIGGER	**Trigger** When ON the module listens and reacts to note-on MIDI events. It changes the frequency of the internal signal based on the currently played MIDI note.
SNAP	**Snap** When ON the module snaps the frequency setting to the nearest semitone.
NEG	**Negate** Negates the internal control signal so it produces a negative frequency instead of a positive (default).
HQ	**High Quality** Enables high quality oversampling.

Phase

Controller
NEW IN V.2

The Phase module lets you control the internal phase control signal by using a single knob. You can think of it as the sidekick to the Frequency module. It offers coarse and fine control over the phase signal, as well at its polarity.

Parameters	
PHASE	**Phase** Coarse phase adjustment. The phase range can optionally be changed to ±1, 2, 4, or 8 periods, through the right-click menu. ±4 periods is the default, which is $\pm 8\pi$rad.
FINE	**Finetune** Fine phase adjustment of ±1 period ($\pm 2\pi$rad). Finetune is not affected by polarity.
MOD	**Phase Mod** Adjust Modulation scale of Phase knob.
POLARITY	**Polarity** Changes phase polarity.
NEG	**Negate** Negates the internal phase control signal.
HQ	**High Quality** Enables high quality oversampling.

FM

Controller

Frequency modulation synthesis (FM synthesis) is a technique for creating complex waveforms with harmonic overtones from simpler waveforms. It works by modulating the frequency or phase of an input signal (Carrier) by another signal (Modulator). For synthesizing harmonic sounds, the Modulator must have a harmonic relationship to the Carrier. However, frequencies that are non-integer multiples of each other (non-harmonic) can also be used, e.g., for creating bells and percussive sounds. The FM module creates a frequency signal that can be used to frequency modulate a generator, by using a simple internal waveform oscillator. The module has support for **Linear FM**, **Exponential FM** and **Phase Modulation**, and has an Advanced Mode that offers Through-Zero, Phase-Init, Antialiasing, DC block, PM Feedback and Exponential Sync.

Parameters	
RATIO	**M/C Ratio** (1/32 to 32/1) Sets the ratio between the two frequencies: Modulator / Carrier.
MODINDEX	**Modulation Index** Tells how much modulation to apply to the signal. The range can be set to 4 (default), 8, 16 or 32 through the right click menu.
FREQ	**Frequency** Finetunes the frequency of the modulator. Right click menu offers range and snap options, and linear frequency option instead of exponential (default).
PHASE	**Phase** Sets the phase of the modulator.
TRIGGER	**Trigger** When ON the module listens and reacts to note-on MIDI events. It changes the frequency control signal based on the received MIDI notes.
HARM	**Harmonic** When ON the module snaps the FM Ratio to integer multiples, in order to produce harmonic frequency modulation.
HQ	**High Quality** Enables high quality oversampling to reduce sideband aliasing.
ADV	**Advanced** When ON expands module to show Advanced parameters (see below).
MODE	**Mode** Choose between **FMLin** (Linear FM), **FMExp** (Exponential FM), **PM** (Phase Modulation), **ModLin** (Linear FM Modulation), **ModExp** (Exponential FM Modulation), **ModPM** (PM Modulation). See **FM Modes** details below.
WAVE	**Modulator Waveform** Choose between: Saw, Square, Triangle, Ramp and Sine (default).

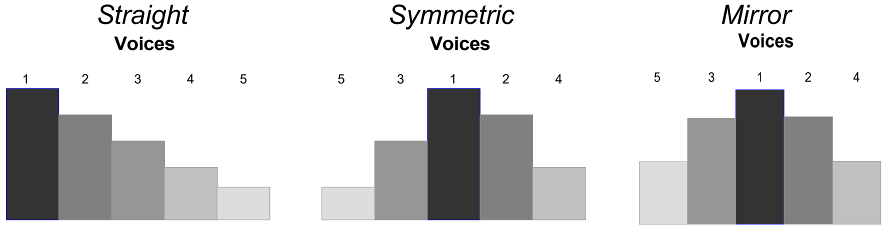
FM Modes	
FMLIN	**Linear FM** Linear FM is a classic synthesis technique where the frequency is swept up and down by a frequency deviation defined by the Modulation Index. Regardless of modulation source, Linear FM always sweeps up the same amount of Hz as it sweeps down. This keeps the output in-tune. Yet, because of its linear nature, large frequency deviations can make the frequency cross 0Hz. This calls for so called *Through-Zero* oscillators that support negative frequencies which output the same signal but in reverse phase. All generators in the synth support Through-Zero, so the module has a **TZ** option to turn Through-Zero and negative frequencies off. With TZ enabled, Linear FM produces output almost identical to Phase Modulation.
FMEXP	**Exponential FM** Exponential FM is a form of frequency modulation where the modulation range follows the musical spacing of notes and octaves. For example, going up one octave means doubling the frequency, and going down one octave means halving the frequency. While this is asymmetrical compared to Linear FM, Exponential FM is in a sense simpler, since this control over tuning frequency is available in most analog synthesizers. At low rates, Exponential FM is simply vibrato, and sounds like Linear FM, while it produces more overtones at high rates. Standard Exponential FM does not have Through-Zero behavior, but instead stops at 0Hz, as we cannot halve a frequency to cross the 0Hz line. However, due to its asymmetrical nature, Exponential FM produces a DC offset that detunes the resulting sound. Therefore, the module features an Exponential Sync (EXS) option to correct for this waveform-dependent detuning (requires Through-Zero (TZ)). DC Block (DC) is recommended for other modulation sources.
PM	**Phase Modulation** Phase Modulation is the original classic FM technique, as invented by John Chowning. Frequency and Phase Modulation are in fact closely related, as modulating a signals phase will also modulate its frequency, simply because frequency is the derivative of phase. PM it therefore also known as *Chowning-style FM* or *indirect FM*, and produces results almost identical to Linear FM, without the need for Through-Zero (TZ), since it only modulates the phase.
MOD LIN/EXP/PM	**Modulation Linear/Exponential/PM** These modes do not actually perform FM/PM, but instead outputs the generated internal modulation signal directly for all modes.

Advanced Parameters	
EXTMOD	**External Mod** Controls the external modulation directly. Modulate this knob for FM/PM modulation with an external signal.
EXTMIX	**External Mix** Controls the mix ratio between the external modulation and the internal.
POLAR	**(Uni-)Polarity** Linearly alters the modulation signal from bipolar to positive unipolar.
FEEDBACK	**Feedback** Sets the amount of feedback. Both FM and PM feedback (PMF) is supported.
TZ	**Through-Zero** Enables Through-Zero FM , meaning that negative frequencies may be generated (default). When OFF, frequencies are clamped to 0Hz. Does not affect PM.
INI	**Phase Init** Enables phase init of the modulation signal, on note trigger. Default is off.
DC	**DC Block** Enables a DC blocking filter on the internal modulation signal.
PMF	**Phase Modulation Feedback** When ON, feedback is PM, when off, FM (default).
EXS	**Exponential FM Sync** When ON, corrects for the detuning of Exponential FM. May require TZ.

Unison

Controller

Unison is a controller for initiating many module instances with slightly detuned frequencies & phase. It can be used to beef up and fatten a sound or to create string-like instruments. Quite similar to polyphony, unison creates extra voices, so to minimize the used CPU resources the module is best placed immediately after a generator to apply unison to. The module supports up to 16 simultaneous detuned voices with volume and pan-spread control. Volume controls the (optionally faded) volume of the detuned voices, while pan-spread spread the voices generated in the stereo panorama. Unison does work with polyphony too. However, a module is internally limited to max 128 voices. This means that it's possible to have, e.g., 16 polyphonic voices with 8 times unison, or 8 polyphonic voices with 16 times unison. But beware, many voices are heavy on the CPU. A warning is displayed if you try to use more than 128 voices.

Parameters	
DETUNE	**Detune Frequency** Detune up to one semitone. The voices will be evenly spread out in the selected frequency range. A **Range** option in the right-click-menu allows up to 4x detune.
PHASE DETUNE	**Phase Detune** Controls the phase detuning of the generated voices (from 0 to 2π).
VOL	**Volume** Control the volume fade of extra voices. A setting of 0 means that the voices fade linearly. A setting of 64 means all voices are output at the same volume (no fade). Finally a setting of –64 means that only voice 1 is output; all other voices have 0 volume. The right-click-menu offers a **Range** option that allows for up to twice the volume of the input, for exaggerating detuned voices.
PAN	**Panspread** Controls how voices are spread in the panorama. Center (0) means no spread.
VOICES	**Voices** Sets number of voices from 1 to 16. Right-click-menu offers a **Constant Power** option (default on), that ensures roughly equal volume regardless of Voices setting.
FREQ/PHASE	**Frequency/Phase Detune** Switch between **Detune Frequency** and **Phase Detune** knobs.
MODE	**Mode** Controls in frequency direction of generated voices: **Straight** (only positive), **Symmetric** (centered around base frequency), **Mirror** (left and right sides are exact duplicates. For even voices this means no fundamental), **Major** (major-scale) or **Minor** (minor-scale), **Oct1x/Oct2x/Oct3x** (straight but offsets successive voices by 1,2 and 3 octaves), **P5/P5Oct1x/P5Oct2x** same but with added perfect fifth. 

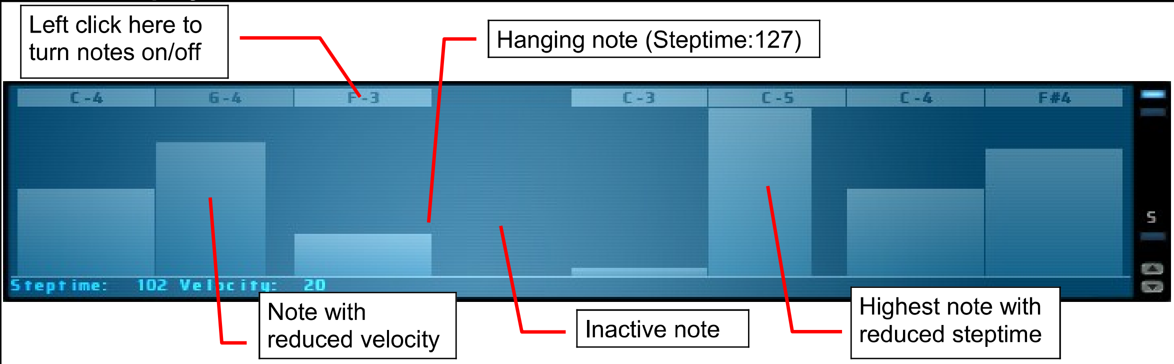
Pattern

Controller

The Pattern module is a small note/velocity step sequencer with a built-in arpeggiator. You can use this controller to play a series of notes (a pattern with 1 to 32 notes) with just one keystroke. Optionally the pattern can loop continuously, and programmed patterns can be stored in 8 available pattern-slots. Like the NotePitch module, *Note*, *Octave* and *Porta* gives control of the base frequency and portamento time (of the internal control signal). The arpeggiator is off by default, but when enabled it will play received notes in a sequence, and optionally changing octaves for each cycle. The buttons to the right of the display switches between pattern display and note masking display. Note masking is a feature that lets you mask off notes in each pattern, thereby forcing the arpeggiator to only play note within a chosen musical-scale.

Parameters	
NOTE	**Note** Transposes the pattern whole semitones up or down. Base-note is shown in the display's text area and in the pattern display. **Tuning** option is available.
OCTAVE	**Octave** Transposes the pattern whole octaves up or down. Base-note is shown in the display's text area and in the pattern display.
PORTA	**Portamento/Glide** Controls how fast the internal frequency signal changes. Full left means no portamento. Right click menu offers two portamento modes: *Fixed* or *Auto*: In Fixed mode glide always takes place. In Auto mode (default), only sustained notes glide.
STEPS	**Steps** (1-32) Sets the number of steps (or notes) in the pattern.
HOLD	**Hold** When on the pattern will continue to play after note-off events.
LOOP	**Loop** When on loops the pattern continuously.
TRIGGER	**Trigger** When on the pattern will start to play on note-on events.
RANGE	**Range** When enabled switches knobs to show controls for range settings (see below).
ARP MODE	**Arpeggiator Mode** **Off** (default): Disables the arpeggiator (only the pattern is played). **Off+Lgto** (Legato): Pattern won't restart as long as at least one key is held. All other settings (AsPlayed, Up, Down,) enables the arpeggiator. Say you hold down C, E, and G on your MIDI keyboard: **Asplayed**: Plays in the same order as received. **Up**: Plays the sequence CEG CEG ... **Down**: Plays the sequence GEC GEC ... **Up/Down**: Plays the sequence CEGEC CEGEC... **Down/Up**: Plays the sequence GECEG GECEG...
PTN	**Pattern** (1-8) Selects the active pattern of the 8 available patterns.
OCT	**Octaves** (1-4) Chooses the number of octaves to show (and use) in the pattern.
ARP	**Arp Octaves** (1-4) Choose the number of octaves to use in the Arpeggiator cycle.
S	**Sustain** When ON, long notes (steptime=127) are held, meaning note-off events are not sent until the next note-on and portamento remains active across note steps.

Pattern Display

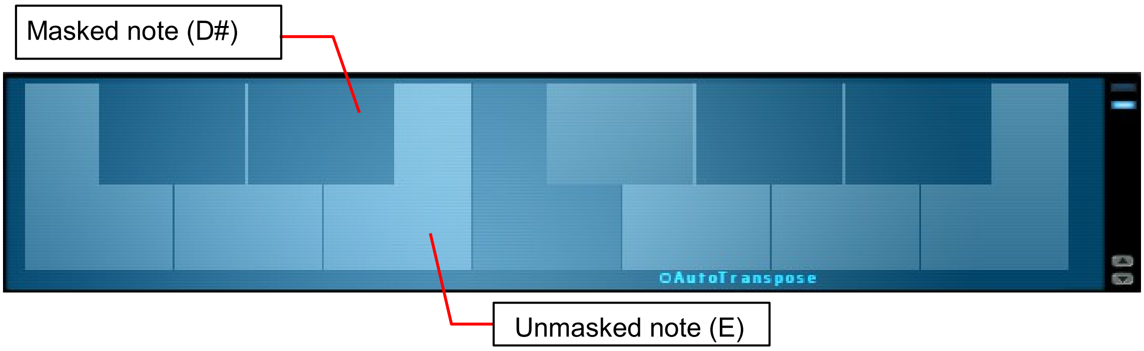


	Left Click on Note Handles Turn notes on or off.
	Left Click on Display Area Adjusts the individual note levels.
	Shift Left Click + Drag Left/Right or Up/Down Left/Right changes Steptime of individual notes. A Steptime of 127 leaves the note sustained, meaning that it does not send note-off event until a new note begins. Up/Down changes Velocity level of individual notes.
	Right Click Shows Pattern Preset Menu.
	Mouse Wheel + Optional Drag Zoom and pan horizontally.
	Shift + Mouse Wheel + Optional Drag Zoom and pan vertically.
	Double Click Mouse Wheel Reset Zoom.

Pattern Buttons

RESET	**Reset** Resets all notes to their default state (note, velocity, and steptime).
RANDOM	**Random** Sets notes to random values.
LEFT	**Left** Shift note pattern left.
RIGHT	**Right** Shift note pattern right.
MIRROR	**Mirror** Flips note pattern horizontally, so the sequence runs backwards.
FLIP	**Flip** Flips note levels (vertically).
SHUFFLE	**Shuffle** Mixes the current notes in a random order without affecting their amplitude.
FILL	**Fill** Copies all the used notes to all the unused steps as a repeated pattern.
TAP	**Tap** Awaits MIDI input, and records each pressed key's note and velocity in the pattern, one step at a time. Tap-recording automatically ends when the pattern is full. To cancel tap-recording, de-click the tap button again.
COPY	**Copy** Copies the current pattern into a copy-buffer. Useful for duplicating patterns.
PASTE	**Paste** Pastes the content of the copy-buffer, overwriting the current pattern.
RECORD	**Record** Similar to Tap, but records incoming MIDI note and velocity with timing data.
TEMPO	**Tempo** Selects the active tempo with respect to the current song tempo (BPM)

Note Mask Display



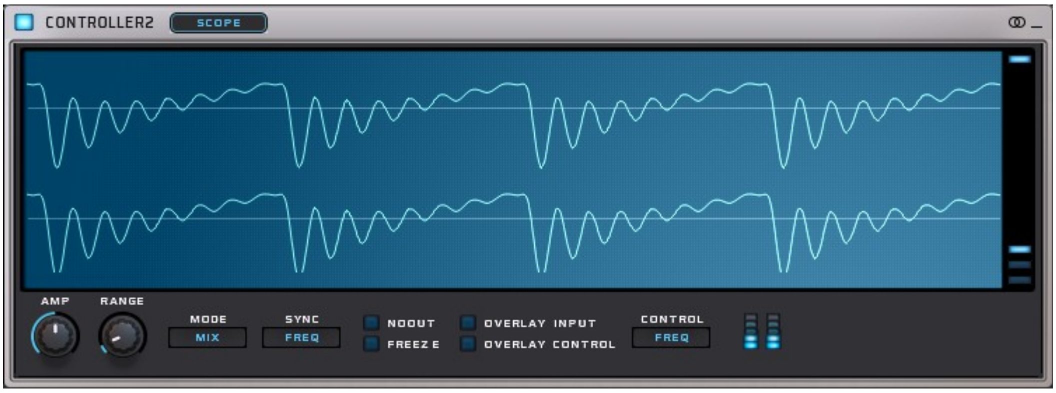
	Left Click on Note Handles Turn notes on or off. Masking a note, means that it will never be played and won't be part of the musical-scale. Hence, when a note is triggered that is masked, it is snapped to the nearest non-masked note. Unmasking a note means that it will play normally, and hence be part of the musical-scale.
	Right Click – Show Right Click Menu Semi +1: Shift masking/scale one semitone to the right. Semi -1: Shift masking/scale one semitone to the left. Scale: Choose between 31 pre-programmed musical-scale presets: *Chromatic, Ionian, Dorian, Phrygian, Lydian, Mixolydian, Aeolian, Locrain, Major Blues, Minor Blues, Diminish, Combination Diminish, Major Pentatonic, Minor Pentatonic, Raga Bhairav, Raga Gamanasrama, Raga Todi, Spanish Scale, Gypsy Scale, Arabian Scale, Egyptian Scale, Hawaiian Scale, Japanese Miyakobushi, Ryukyu Scale, Wholetone, Minor 3rd Interval, 3rd Interval, 4th Interval, 5th Interval, Octave Interval*
AUTO-TRANSPOSE	**AutoTranspose** When on, automatically transposes the scale with respect to the *Note* knob.

Range Parameters

STEP	**Steptime** Sets the steptime limit for notes. A setting of 127 means no limit (default), where keys are allowed to be sustained, without any note-off.
VELOCITY	**Velocity** Sets the velocity scale (default is 127: no scale).
MIN	**Min Note** Sets the minimum allowed MIDI note to be played from 0 to 255. The right-click-menu offers a Limit setting on how to handle notes outside range: **Clamp** (clamps to last allowed note), **Wrap** (wraps within the last octave), **Mirror** (mirrors note within the last octave), **Mute** (mutes the note). Default is *clamp*.
MAX	**Max Note** Set the maximum allowed MIDI note to be played from 0 to 255. **Limit** menu is available. See *Min*.

Scope

Controller
NEW IN V.2

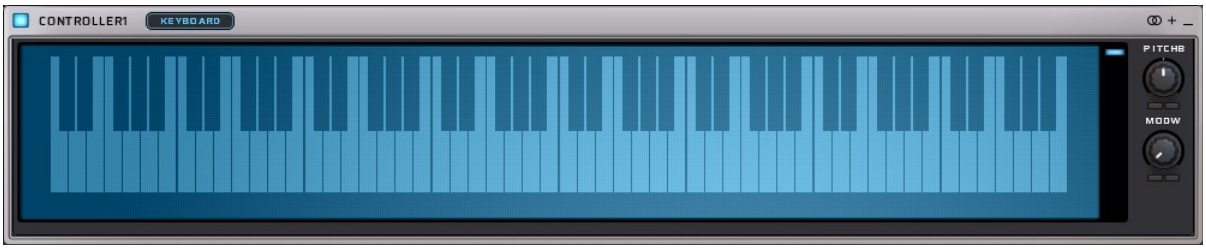


The Scope module its an advanced waveform oscilloscope. It offers analysis of up to two input signals simultaneously. Unlike any other module, the Scope module does not require an output. This means that you can route a signal from any module into a Scope in order to inspect its signal, without hooking the Scope up to the Output module or the signal chain. However, if the Scope is in the signal chain, it offers added analysis of the control signals and MIDI data, since it syncs the scope to the frequency signal. The scope can also sync to the song tempo (BPM), trigger signal or be left free running. The Mode selection determines if input to the Scope should be mixed before analysis or inspected separately. This is useful if you want to view two signals overlayed, simultaneously, using the *overlay input* option. In addition the Scope offers an *overlay control* option, that shows the internal control-signals in the synth. Finally the Scope offers frequency display and phase display. The Scope is only for inspection, which means that the output from the Scope module (if any) is always the same as its input.

Parameters	
AMP	**Amp** Amplifies the input signal for improved inspection.
RANGE	**Range** Sets the time-based range of the scope. A low range means that you zoom-in to the signal, where as a high range means that you zoom-out.
MODE	**Mode** Display Mode. Switch between **Mix** (Mixes all inputs before display), **Input1** (Shows first input), **Input2** (Shows second input), **Control** (Shows control signal).
SYNC	**Sync** Determines the scopes range synchronization. Choose between, **Freq** (Sync range to MIDI input or control signal frequency if available), **BPM** (Sync range to song tempo), **Trigger** (Sync by restarting scope on Note-on events), **None** (free running).
NOOUT	**No Output** Disables output from Scope. This is useful if you, e.g., want to inspect a control signal, but don't want the Scope to contribute with any sound.
FREEZE	**Freeze** Temporarily freezes the Scope. The Freeze LED will blink to indicate that the display is frozen. Click again to un-freeze.
OVERLAY INPUT	**Overlay Input** When on and having two input signals, this overlays the other input signal with the input signal.
OVERLAY CONTROL	**Overlay Control** When on, overlays the chosen control signal with the input signal.
CONTROL	**Control** Choose the control signal to display: **Freq** (Frequency control signal), **Phase** (Phase control signal), **Gate** (Note-on and Note-off MIDI trigger control signal), **Note** (MIDI Note control signal).

Keyboard

Controller
NEW IN V.2



The Keyboard module is a MIDI-processing and visualization module. It highlights the incoming MIDI-notes visually on a virtual keyboard, for both monophonic and polyphonic MIDI signals. It is normally best used to for visualizing incoming MIDI by hooking it up at the end of the signal chain, just before the Output module. That way it is easy to monitor exactly what notes are being triggered. It can also be used to visualize the internal MIDI. For instance, hooking it up just before a Pattern module, will make it display the interesting note-patterns generated by the Pattern module's arpeggiator. A small keyboard view (8 octave) is the default. A larger keyboard view (10 octave) can be toggled via the panel switch. Aside from pure visualization, the module can also be used to trigger internal MIDI. Left clicking on the virtual keyboard with the mouse will produce MIDI Note-On and Note-Off events. Moreover, a Pitchbend knob and a ModWheel knob will generate Pitchbend and ModWheel MIDI control events respectively. The module does not produce or alter audio in any way, meaning that all audio input is simply passed through.

Keyboard Display	
	Left Click Keys Left clicking on the virtual keyboard with the mouse will produce MIDI Note-On and Note-Off events. Default range is 8 octaves, spanning from lowest octave, starting with C0, to highest octave, starting with C7. Toggling expanded mode via the panel switch yields a 10 octave keyboard, spanning octave C-1 through to octave C8.
	Shift + Left Click Keys Holding Shift while mouse clicking will make keys sticky, and hence only send MIDI Note-On events until pressed again.

Parameters	
PITCHB	**Pitchbend** Sends Pitchbend MIDI control events (0xe0). The knob automatically snaps to center position as you would expect from a regular pitchbend wheel.
MODW	**Modwheel** Sends Modwheel MIDI control events (0xb0).

Delay

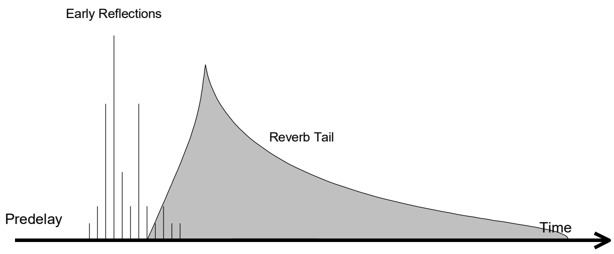
Effect

The Delay module is a tempo controlled stereo delay, featuring two independent delay lines. The delay time can be set separately for the left and right channels using the Stereo knob. A coarse delay parameter that sets the delay time in eights of a beats (0-16 beats/8) and a fine delay parameter that adjusts the delay steplessly. How much of the delayed signal that is sent back to the delay is controlled by the Feedback knob, and can optionally be low/high pass-filtered. A Tap parameter controls the volume of the first delay tap for both channels. A built-in adjustable cross delay feedback mixer makes it possible to achieve the popular stereo effect known as *ping pong*, where a part of the feedback from the left channel is sent to the right channels input and vice versa.

Parameters	
DELAY	**Delay** (Beats) Sets the delay in beat, snapping to one eighth (1/8) of a beat.
FINE	**Fine** (ms) Finetunes the delay steplessly up or down one beat. A **Range** option allows the fine range to be tuned to beat-fractions of 1/8 (default), 1/4, 1/2 and 1.
FBACK	**Feedback** Sets the amount of feedback, i.e., how much of the already delayed signal to send back into the delay.
STEREO	**Stereo** Offsets the delay in the right channel by plus/minus one beat compared to the left.
TAP	**Tap** Controls the volume of the input signal to feed to the delay.
PINGPONG	**Pingpong** When in stereo, the feedback can work in two modes. Either the signal is sent back to the sample channel (i.e., left to left and right to right) or alternatively to the other channel (i.e., left to right and right to left). When the channel delays are offset the latter can give an interesting "ping pong" effect between the left and right speaker. This knob controls how much ping pong you want. A setting of 0 means no ping pong. 127 means full ping pong, and centered means 50% of both.
LP	**Lowpass** (Hz) Lowpass cutoff knob for filtering the feedback. At 127 all frequencies pass.
HP	**Highpass** (Hz) Highpass cutoff knob for filtering the feedback. At 0 all frequencies pass.
DRYOUT	**Dry Out** Enables dry output, meaning that the original signal is mixed with the delayed signal. When disabled only the delayed signal is output.
FILTER	**Filter** Enables feedback filtering. This has to be enabled for LP and HP to work.
ANTIPHASE	**Antiphase** When ON, the fine delay is added in reverse phase to left and right channels.

Reverb

Effect



The Reverb module adds reverberation to the incoming signal, giving the impression of sound being played inside a room or other confined space.

Parameters	
DECAY	**Decay** (ms) Controls how fast the reverb tail dies out (fades) from 0 to 10 seconds.
SIZE	**Size (Diffusion)** Controls the virtual size of simulated room from small, with few and fast wall-to-wall reflections, to large, with many diffusely spaced out reflections.
DRY	**Dry** (dB) Sets the Dry amount, i.e., how much of the original signal to mix with the output.
WET	**Wet** (dB) Sets the Wet amount, i.e., how much of the reverb signal to mix with the output.
PRE	**Predelay** (ms) Sets the pre-delay: The time delay before the reverb kicks in (0 to 500ms).
FBACK	**Feedback** Sets how much of the reverb signal to feed back into the reverb.
DAMP	**Damp** Controls the attenuation (damping) of the high frequencies in the reverb tail.
PAN	**Pan** Pans the reverb signal left or right.
LP	**Lowpass** (Hz) Lowpass cutoff knob for filtering the reverb tail. At 127 all frequencies pass.
LP GAIN	**Lowpass Gain** (dB) Sets how much of the lowpass signal to blend in.
HP	**Highpass** (Hz) Highpass cutoff knob for filtering the reverb tail. At 0 all frequencies pass.
HP GAIN	**Highpass Gain** (dB) Sets how much of the highpass signal to blend in.
BEAT SYNC	**Beat Sync** When on predelay is forced to be in eighth beats (1/8). When off it is stepless.
FILTER	**Filter** Enables reverb tail filtering. This has to be enabled for LP and HP to work.
MOD	**Modulation** Switches modulation on or off. This is very slow time varying delay modulation that smears out the reverb, making it softer.
STEREO	**Stereo** When on, two separate stereo tails are mixed, one for the left and the right input. In mono mode, only one stereo tail is generated using a mono version of the input.
REVERB TYPE	**Reverb Type** Chooses the algorithm to use for reverb simulation, from simple to complex. The reverbs are named according to they sound and properties: Hall, SmallRoom, Ambience, BigHall, Live, DrumBox, Cathedral and Megaverb.
ER	**Early Reflection** (dB) Sets the level in dB of the early reflections. These are very short spaced out echoes that appear immediately after the predelay, but before the reverb tail.

Distortion

Effect

The Distortion module does Waveshaping, Resampling and Bit Reduction. Waveshaping changes the shape of the incoming waveform using a nonlinear transfer function. Resampling is a digital effect that takes an incoming signal and picks out samples at a specified interval in order to simulate an (in this case) lower sampling rate. Similarly, bit reduction (bit crushing), squashes the incoming samples dynamic range to simulate less bit precision.

Parameters	
AMOUNT	**Amount** Sets how much of the distorted signal you wish to hear by adjusting the mix ratio between the incoming- and the distorted-signal. 127 means full distortion.
DRIVE	**Drive** Controls the internal variable(s) in the waveshaping algorithm. Cranking it up yields a more distorted sound.
BITS	**Bits** Lowering this reduces the bit precision. When set to max no reduction is performed. The right-click menu offers a **Stereo** option that makes the Bit reduction run in antiphase (off by default).
RATE	**Rate** Controls the sample rate conversion from (resampling). When set to max no resampling is performed. The right-click menu offers a **Stereo** option that makes the Rate reduction run in antiphase (off by default).
HQ	**High Quality** Toggles oversampling to reduce high frequency aliasing.
TYPE	**Distortion Type** Chooses the Waveshaping algorithm: Soft, Hard, Variable, Soft-fuzz, Hard-fuzz, Soft-tube, Hard-tube, Warmer.

Phaser

Effect

Phaser is an effect module for simulating electronic stereo phasing: A type of filtering that alters the phases of the different frequency components in the signal using 1 pole allpass filters. Each allpass filter generates a phase shift of 180 degrees, producing one notch for every two stages. The left and right phase shifts are usually modulated by, e.g., a LFO to create a characteristic swooshing sound. Moreover, the phased signal can be fed back to the phaser to intensify its effect.

Parameters	
LDELAY	**Left Delay** Sets the phase shift delay for the left channel.
RDELAY	**Right Delay** Sets the phase shift delay for the right channel.
AMOUNT	**Amount** (dB) Sets how much of the phased signal you wish to hear by adjusting the mix ratio between the incoming- and the phased-signal. At a value of 0, only the input signal is audible, at a value of 127, only the phased signal is audible.
FEEDBACK	**Feedback** Controls how much of the phased signal to send back into the phaser.
NEG FBACK	**Negate Feedback** When ON shifts the polarity of the feedback.
ANTIPHASE	**Anti Phase** When ON flips the direction of Rdelay knob, so the left and right channel run out of phase. Useful for creating interesting stereo phasing with a single LFO.
TYPE	**Phaser Type** Selects the number of phaser stages (or allpass filter poles), 1-8.

Chorus / Flange

Effect

Chorus is a stereo effect for simulating several simultaneous voices. It does this by introducing a delay (up to 200ms) and varying this delay over time using a modulator. The modulation results in a slightly detuned signal that combined with the input signal sounds like an extra voice. Filtered feedback can be used to intensify this effect. If the delay is less than around 20ms, then the effect is called *flange*: A comb filter effect, where notches appear in the frequency spectrum. Varying the time delay causes these to sweep up and down the frequency spectrum, resulting in a swooping sound.

Parameters	
LDELAY	**Left Delay** Sets the relative delay for the left channel.
RDELAY	**Right Delay** Sets the relative delay for the right channel.
AMOUNT	**Amount** Adjusts the mix ratio between the input and the chorus-signal. Right-click-menu offers a **Constant Power** option (default on), that ensures roughly equal volume regardless of Type setting.
FEEDBACK	**Feedback** Controls how much of the output signal to send back into the chorus.
DEPTH	**Depth** Controls the overall depth of the delay.
SPREAD	**Spread** Offsets the delays and phases between the different chorus stages.
RATE	**LFO Rate (0-100Hz)** Sets the internal LFO rate in Hz.
AMP	**LFO Amp** Set the internal LFO amplitude.
XFEED	**Stereo Crossfeed** Sets the amount of feedback crossfeed from left to right, and right to left channel.
WIDTH	**Stereo Width** Expands or contracts the stereo width of the Chorus signal.
LP	**Lowpass** Lowpass cutoff knob for filtering the feedback. At 127 all frequencies pass.
HP	**Highpass** Highpass cutoff knob for filtering the feedback. At 0 all frequencies pass.
SERIES	**Series** Run the chorus stages in series, or in parallel (default).

Parameters (Chorus, continued)	
FLT CH	**Filter Chorus** Apply LP and HP filter to Chorus-signal path.
FLT FB	**Filter Feedback** Apply LP and HP filter to Feedback-signal path.
ANTIPHASE	**Anti Phase** When ON flips the direction of right delay, so left & right channel run out of phase.
NEG FB	**Negate Feedback** When ON shifts the polarity of the feedback to negative.
ADV	**Advanced Mode** Exposes XFeed, Width, LP/HP and Feedback Delay parameters.
WAVEFORM	**LFO Waveform** Sets the internal LFO waveform: Sine, Triangle (default), Square, Saw.
TYPE	**Chorus Type** Selects the number of chorus stages, 1-8. More voices means fatter chorus.
PHASE	**Phase Offset** Sets the modulation signals phase offsets to fatten up the chorus.
FBD	**Feedback Delay** Sets the internal pre-delay before feedback signal is fed back.

Folder

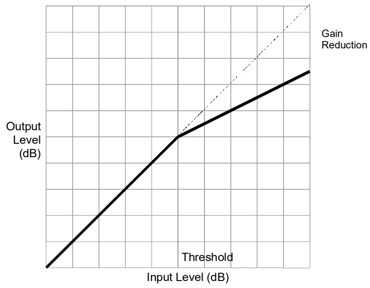
Effect
NEW IN V.2

The Folder is a classic Sine based waveshaper module. Technically, the Folder linearly frequency modulates a 0Hz Sine carrier oscillator with the input as modulation signal. Due to the curved nature of the sine wave, this means that the input is folded based on its amplitude. The *Folds* knob controls the drive of the folding, where one stage of folding ranges from almost no folding [-½π;+½π] to full folding [-π;+π]. Several folding stages can be further processed in series. The *Symmetry* knob adds or subtracts a DC offset to the input signal before folding, giving control over the symmetry of the folding process. Finally, the *Feedback* knob allows for both positive and negative feedback.

Parameters	
AMOUNT	**Amount** Sets how much of the folding signal you wish to output by adjusting the mix ratio between the incoming- and the outgoing-signal. 127 means full folding.
FOLDS	**Folds** Sets the amount of folding drive, ranging from almost no folding [-½π;+½π] to full folding [-π;+π].
SYMMETRY	**Symmetry** Adds a positive or negative DC offset to the input signal before folding. The right click menu offers a **Stereo** option which flips the offset for the right channel.
FBACK	**Feedback** The output is fed back into the input, multiplied by this feedback factor.
SOFT / HARD	**Soft / Hard** Controls the overall drive of the folding. **Soft** scales the input by ½ before folding, and scales the output by 2 after folding. **Hard** does not rescale the input, but still scales the output by 2.
HQ	**High Quality** Enables high quality oversampling.
STAGES	**Stages** Sets the amount of folding stages in series (1-4). 1 is default.

Compressor

Effect



The Compressor module is a soft-knee compressor with Peak/RMS detection and optional sidechain. This may sound intimidating, but a compressor is really quite simply a time varying amplifier that makes a signal appear louder by changing the output volume based on the input volume. By using an envelope follower it reduces the gain of an incoming signal if its amplitude exceeds a certain threshold, where the amount of gain reduction is determined by a ratio control. Typically the result is then amplified again (*makeup gain'ed*) in order to fully utilize the available dynamic range. The final result thus appears louder since the low amplitude part is expanded while the top part is compressed.

Parameters	
ATTACK	**Attack** (2-500ms) The attack time for the envelope follower used to determine the amplitude level.
RELEASE	**Release** (2-1000ms) The release time for the envelope follower used to determine the amplitude level.
RATIO	**Ratio** (1.05:1 – 20:1) Controls how much compression should be applied when the signal level exceeds the threshold. A gain reduction ratio of N:1 means that a level increase in the input signal of N dB results in level increase in the output signal of 1 dB.
THRESHOLD	**Threshold Level** (dB) The threshold is a value in dB where the compressor will kick in. The signal part below the threshold will not be compressed.
KNEE	**Knee** (dB) Instead of the kinked compression curve shown above (hard knee), the module can soften the curve (soft knee). *Knee* sets the distance from the threshold value where the knee curve starts. Lowest value means hard knee compression. The higher the value the larger and softer the knee curve will be.
GAIN	**(Makeup) Gain** (dB) The make up gain raises the entire signal after compression to make use of the extra headroom / dynamic range given by the compression.
SIDECHAIN	**Sidechain** Inputs another signal or offset to the envelope detector, meaning that you can controls the gain reduction of one signal from the amplitude of another signal (the sidechain). When centered the sidechain signals equals zero.
MIX	**Mix** Controls how to mix the sidechain with the module's input. A setting of 0, means 100% input. A setting of 127 means 100% sidechain.
PEAK / RMS	**Peak / RMS (Root Mean Square)** Selects whether the envelope follower should be based on Peak or RMS detection. *Peak* is usually best suited for heavy, fast compression (percussion), while *RMS* is good for subtle, musical compression.
MONITOR	**Monitor** Turns ON gain reduction monitoring for this module instead of the standard waveform and frequency display. Useful for verifying that the module actually does compress the sound

MultiComp

Effect
NEW IN V.2

The MultiComp module is a multiband soft-knee compressor width adjustable crossover frequencies and Peak/RMS detection. You can think of this module as three instances of the Compressor module, each operating on a single band of a three band EQ with Low, Medium, and High outputs. For a smooth overall frequency and phase response, the band separation is done using State Variable Filters with a Linkwitz-Riley crossover design. Each compressor have their own separate options, and each band can either run through its dedicated compressor (default), be bypassed (no compression) or muted. The attenuation in dB is shown graphically for each band. Each band can also be temporarily solo'ed. For stereo input, the left and right channel-compression can either be Linked or run in stereo (unlinked).

Parameters	
LOW	**Low** (-12dB to +12dB) Boost or cut the Low-band frequency range.
MID	**Mid** (-12dB to +12dB) Boost or cut the Mid-band frequency range.
HIGH	**High** (-12dB to +12dB) Boost or cut the High-band frequency range.
AMOUNT	**Amount** Sets the mix ratio between the unprocessed (dry) and the compressed signal (wet). Useful for parallel compression. A setting of 127 means full compression.
PEAK / RMS	**Peak / RMS (Root Mean Square)** Selects whether the envelope follower should be based on Peak or RMS detection. *Peak* is usually best suited for heavy, fast compression (percussion), while *RMS* is good for subtle, musical compression.
LINK	**Link L/R** Stereo link left and right channels. When on, the compression in both channels is exactly the same, and based on the maximum of the Left and Right channels. When off the compression is done separately for each channel.
SOLO	**Solo** Temporarily solo the currently selected band. The Solo LED will blink to indicate that the current compressor/band is soloed. Click again to un-solo.
BAND	**Band** Switch between Low, Mid and High band selection. Since each band has separate compression parameters, this alters the compression knobs. The selection can also be done from the band display.
FREQ1	**Freq1** (50Hz to 550Hz) Sets the crossover frequency between Low and Mid bands.
FREQ2	**Freq2** (600Hz to 5kHz) Sets the crossover frequency between the Mid and High bands.
GAIN	**Final Gain** (-12dB to +12dB) Controls the final gain compensation after compression but before dry/wet mixing.

Per Band Compression Parameters	
ATTACK	**Attack** (0-500ms) The attack time for the envelope follower used to determine the amplitude level.
RELEASE	**Release** (0-1000ms) The release time for the envelope follower used to determine the amplitude level.
RATIO	**Ratio** (1.05:1 – 20:1) Controls how much compression should be applied when the signal level exceeds the threshold. A gain reduction ratio of N:1 means that a level increase in the input signal of N dB results in level increase in the output signal of 1 dB.
THRESHOLD	**Threshold Level** (-30dB to 0dB) The threshold is a value in dB where the compressor will kick in. The signal part below the threshold will not be compressed.
KNEE	**Knee** (1 to 20dB) Controls the softness of the compression curve. *Knee* sets the distance from the threshold value where the knee curve starts. Lowest value means hard knee compression. Higher values means soft-knee compression.
MAKEUP	**Makeup** (0 to 12dB) The make up gain raises the entire signal after compression to make use of the extra headroom / dynamic range given by the compression. For normal operation, it is recommended to keep this set at 0 and use the Low, Mid and High knobs for gain adjustment.
COMP / BYPASS / MUTE	**Comp / Bypass / Mute** Per Band selection: Each band can either run through its dedicated compressor (default), be bypassed (no compression) or muted.

PitchShift

Effect

The PitchShift module changes the pitch of an incoming signal steplessly up or down one octave. This is an effect and thus not related to pitch changes in any of the generators or controllers, but works on all incoming signals. It does its magic by using two delay lines with time varying delays. The delays are filled at a fixed rate and read back at a variable rate, causing the pitch shift. To avoid discontinuities the two delay lines work out of sync, so when one increases its delay the other decreases its delay. The two delays are then amplitude modulated and mixed in order to sound like a continuous pitch shift.

Parameters	
SHIFT	**Shift** The amount of pitch shifting to apply. Centered (64) means no pitch shifting. 127 means up one octave, while 0 means down one octave.
DEPTH	**Depth** The amplitude modulation time, or how long to use the result from one delay line.
AMOUNT	**Amount** Sets how much of the pitchshifted signal you wish to hear. 0 means no effect. 127 means full pitchshifting.
FEEDBACK	**Feedback** Since this is a delay based effect, feedback can be applied. *Feedback* controls how much of the output signal to send back in to the pitchshifter.

BitShuffle

Effect

The BitShuffle module is a digital distortion effect, inspired by a common hardware circuit-bending technique. It reduces an incoming signal to between 1 and 16 bits, and allows you to shuffle and choose from the available bits to reconstruct the output. The module also does simple dithering (1 bit noise) to reduce bit reduction artifacts. It can be used to do distortion ranging from very subtle (many bits, little shuffle) to extremely hard distortion (few bits, lots of shuffle).

Parameters	
BITS	**Bits (1-16)** Choose the number of bits for the internal digital representation. 1 bit being the lowest quality (sound can only be on or off) to 16 bits (CD quality).
AMOUNT	**Amount** Sets how much of the bitshuffled signal you wish to hear. 0 means no effect. 127 means full bitshuffle.
GAIN	**Gain** Since bit-shuffling can affect the overall volume level, this parameter enables you to change the volume up or down after bitshuffle is applied.
DITHER	**Dither** Controls the amount of dither to add to the signal. Dither is simply one bit noise added to the signal before the bit reduction quantization. 0 means no dither, 127 means full dither.

BitShuffle Buttons	
	
RESET	**Reset** Resets all bits to their default ordering (decreasing).
RANDOM	**Shuffle** Shuffles the selected bits in a random order.
LEFT	**Left** Shift/Rotate bits left.
RIGHT	**Right** Shift/Rotate bits right.
MIRROR	**Mirror** Flips bit-pattern in reverse order.
RANDOM	**Random** Set bit values to random numbers (between 1 and the selected number of bits)

PanSpread

Effect

The PanSpread module is a stereo expander with individual low and high spread, plus crossover frequency adjustment. It is useful for widening the stereo image of a stereo input signal. The module has a built-in filter that separates low frequencies from high, as you normally want different stereo responses at low and high frequencies. The Low and Hi parameters offers extreme widening up to 400%. Yet, care should be taken not to over-do stereo widening, as this can lead to stereo phase cancellations in the final mix. If the input signal is mono, the module has no effect. If you want to introduce stereo to a mono-only signal, one trick is instead to use a Delay module with a fine delays of <1ms to offset the left and right channels slightly.

Parameters	
AMOUNT	**Amount** Sets how much of the panspread signal you wish to output by adjusting the mix ratio between the incoming- and the outgoing-signal. 127 means full panspread.
FREQ	**Frequency** (20Hz-20kHz) Sets the crossover frequency of the internal lowpass and highpass filter.
LOW	**Low** (0%-400%) Widens the Low frequency signal by 0% (mono) to 400%. Center means 100% (no change).
HI	**High** (0%-400%) Widens the High frequency signal by 0% (mono) to 400%. Center means 100% (no change).
MODE	**Mode** Choose between modes: **Mix** (default), **Low Only** (only output low part of the filtered signal) and **Hi Only** (only output high part of the filtered signal).

Maximizer

Effect

The Maximizer module is a Sonic Maximizer emulation with Low Contour, Process and Peak Detector. The Sonic Maximizer is a classic sound enhancer effect, that does its magic by splitting the sound into multiple bands and phase offsets each band individually. This typically results in a boost in treble and bass. To counter the harsh nature of high frequency transients in the treble, the peak detector can attenuate peaks.

Parameters	
AMOUNT	**Amount** Sets how much of the maximizer signal you wish to output by adjusting the mix ratio between the incoming- and the outgoing-signal. 127 means full maximizer.
LOWC	**Low Contour** Sets the amount of phase corrected bass frequencies.
PROCESS	**Process** Sets the amount of phase corrected treble frequencies.
PEAK	**Peak Detector** Sets the amount of peak detection, to keep peaks in check. Higher values mean more peak detection and attenuation. A value of 0 means no peak detection.

Limiter

Effect
NEW IN V.2

The Limiter module is a simple soft-knee, peak limiter with optional stereo link option. A limiter is in many ways similar to a compressor, except that it offers no ratio and makeup controls. It is especially useful for ducking peaks that cross the limiting threshold, thus keeping them in check and making sure they don't clip. However, due to the synth's strict low latency requirements the Limiter offers no look-ahead option, which means that the output may still exceed the threshold limit, depending on the attack and release settings.

Parameters	
ATTACK	**Attack** (ms) The attack time for the envelope follower used to determine the amplitude level.
RELEASE	**Release** (ms) The release time for the envelope follower used to determine the amplitude level.
THRES	**Threshold** (-20dB to 0dB) Sets the threshold in dB where the limiter kicks in.
KNEE	**Knee** Sets the softness of the limiting. 0 means hard limiting, higher values soft limiting.
LINK	**Link L/R** Stereo link left and right channels. When on, the limiting in both channels is exactly the same, and based on the maximum peak of the left and right channels. When off the limiting is done separately for each channel.
MONITOR	**Monitor** Turns on gain reduction monitoring for this module instead of the standard waveform and frequency display.

SoftClip

Effect
NEW IN V.2

The SoftClip module is a soft clipping distortion effect. It works by soft clipping the peak parts of the amplitude signal above a desired threshold. The *Soft* knob can be used to control the level of softness, ranging from hard clipping to -6dB below-threshold soft-clipping. The module can automatically account for the loss in gain based on the threshold setting, and can optionally add digital distortion to the soft clipped signal, featuring adjustable stereo bit-reduction and noise.

Parameters	
THRES	**Threshold** Sets the threshold level in dB for clipping to occur.
SOFT	**Soft** Sets the amount of soft clipping, ranging from 0dB (hard clipping) to -6dB (soft-clipping).
DRIVE	**Drive** (-20dB to +6dB) Optionally scales the input signal before clipping.
AMOUNT	**Amount** Adjusts the mix ratio between the incoming- and the soft-clipped-signal. A value of 0 means no-clipping,while a value of 127, means 100% soft-clipping.
DIST	**Distortion Amount** Sets the amount of digital distortion to add to the soft-clipped signal.
BITS	**Bits** Adjusts the digital bit-reduction of the clipped signal. When set to max no reduction is performed. The right-click menu offers a **Stereo** option that makes the Bit reduction run in antiphase (on by default).
BIAS	**Bias** Controls the negative and positive stereo-offset bias of the Bit reduction.
NOISE	**Noise** Amount of bit-scaled noise to add after bit reduction.
AUTOGAIN	**Auto Gain** When enabled, automatically sets the gain based on the clip threshold level.
DIST	**Digital Distortion** This enables digital-distortion (bit reduction) on the soft-clipped part of the signal, controlled by the Dist, Bits, Bias and Noise parameters.
HQ	**High Quality** Enables high quality oversampling.
DIFF	**Difference** Outputs the difference between the clipped an unclipped signal.

Repeater

Effect
NEW IN V.2

The Repeater module is a sample-based delay effect. On a note-on event, it repeats the input once a certain input interval has been buffered in a delay-line. It then replays the content of this buffer until it receives a note-off event, or the buffer has been repeated a fixed number of times defined by the *Repeat* knob. Optionally the replay speed of the buffer can tweaked using the *Delta* knob. A Sync option is available to sync the interval to the host tempo.

Parameters	
INTERVAL	**Amount** (ms) Sets the repeat-buffer's timing-interval. When sync is on this interval is set in fractions of a beat.
REPEAT	**Repeat** (0-31) Sets the number of times the buffer should be repeated. 127 (default) is AUTO mode, which means that it repeats infinitely until a note-off event is received.
DELTA	**Delta** Tweaks the playback speed of the delay buffer, either positively (faster) or negatively (slower).
AMOUNT	**Amount** Sets how much of the repeated signal you wish to output by adjusting the mix ratio between the incoming- and the outgoing-signal. 127 means full repeater.
SYNC	**Sync** When on, repeat-buffer's interval is synced to host tempo.

SubPatch

Effect
NEW IN V.2

The SubPatch module is unique multi-functional module available as both Generator and Effect. A SubPatch is essentially a container for another patch. The module therefore features a set of programmable control-knobs, that can be used to control the patch from the parent-patch. The SubPatch can be uses as a module-container for tidying up patches, and for reusing already created (sub)patches inside other patches. The synth supports up to two nested levels of subpatches. See also section 6.5 (p.46).

Parameters	
CONTROL	**Control** When control is enabled, the frequency and phase control signals received by the module will be passed on to the patch in the SubPatch. When off, only MIDI events received by the module are passed-on.
EDIT	**Edit** When on, shows the Edit buttons. When off, hides the Edit buttons.
ADV	**Advanced** Enables advanced mode. This expands the SubPatch module to show 8 programmable control-knobs instead of just 4.
PATCH	**Patch** Dropdown menu for selecting the active patch in the SubPatch. There is room for up to 8 patches in each SubPatch.

Edit Buttons	
	
EDIT	**Edit** Edits the currently selected patch. This brings the selected patch into view for editing. To return to the parent patch, locate and click the button in the Control Panel that says **Exit Subpatch**.
LOAD	**Load** Opens the patch browser for directly loading an existing patch into the currently selected patch slot.
RENAME	**Rename** Renames the currently selected patch.
CLEAR	**Clear** Clears the current patch. Resetting it to default.
ADD	**Add** Adds a new patch to the subpatch, and switches to it.
REMOVE	**Remove** Removes the currently selected patch from the subpatch module.

LFO

Modulator

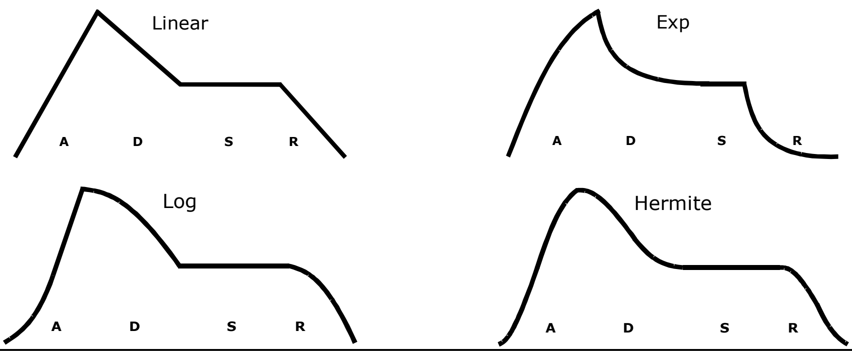
The LFO module is a simulated analog *Low Frequency Oscillator* that generates Sine, Triangle, Square, Saw and Random waveforms at frequencies up to 100Hz. A LFO is typically used for modulating other parameters, but in some cases can also be used as replacement for a standard low frequency generator.

Parameters	
RATE	**Rate** (0-100Hz) Sets the frequency of the oscillator.
AMOUNT	**Amount** Sets the maximum amplitude of the waveform, i.e., how "large" the waveform is.
FADE	**Fade** (0-100ms) Sets an optional fade-in time, in which the oscillator will slowly fade the amplitude from zero to max on trigger events.
PHASE	**Phase** Controls the phase of the oscillator, i.e., where in the waveform cycle are we ?.An optional **Random Poly Phase** option is available through the knob's right click menu, adding a randomized Phase Init offset when triggered polyphonically.
KYBD	**Keyboard Tracking** Sets the amount of keyboard tracking from –200% to 200%. This offsets the LFO frequency with respect to the incoming frequency control signal. Setting this to 100%, means that the LFO frequency will track the played note frequencies exactly with respect to the selected **Keyboard Tracking Base Key**. The **Base Key** can be setup through the knobs right click menu (default:C1).
BIPOLAR	**Bipolar** When ON the module produces a signal in the amplitude range [–1;1]. When OFF it only uses the amplitude range [-1;0] (Unipolar).
INV	**Invert** Inverts (or flips) the oscillator signal. Useful for say outputting in the [0;1] range in unipolar mode, or to simply generate inverted waveforms.
TRIGGER	**Trigger** When ON the module restarts the waveform on MIDI Note-on events. The waveform is restarted from the currently set Phase value. Fade is also activated if Fade is set to anything above 0.
SYNC	**Sync** Synchronizes the oscillator rate to a whole number of beats. Useful for running modulation in sync to a songs tempo.
KYBD/FADE	**Kybd/Fade Toggle** Toggles between showing the **Kybd** and **Fade** parameter knobs.
WAVEFORM	**Waveform** Selects the desired oscillator waveform. Choose between: Sine, Triangle, Square, Saw, Random and Random16. Random16 has 16 step changes per period, unlike Random which only has one.

ADSR

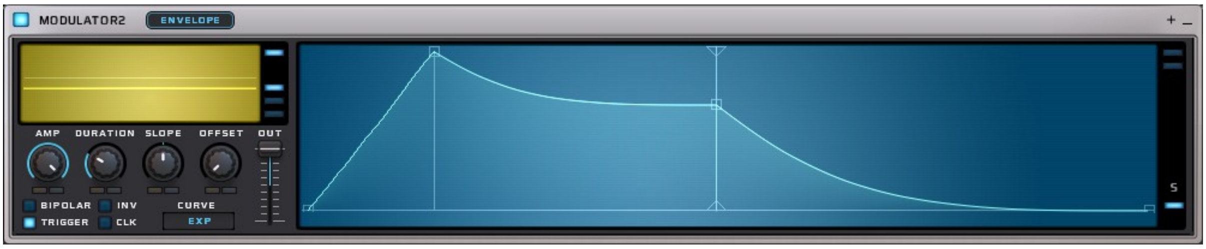
Modulator

ADSR module is a four-stage envelope consisting of *Attack, Decay, Sustain* and *Release*.

Parameters	
A	**Attack** (0-10000ms) Attack is the time taken for the initial level to go from from 0 to 100%.The module defaults to a forced reset to 0 on each retrigger. An alternative **Analog** attack option is available through the right click menu, where a retrigger instead rises from the currently reached envelope level. Attack also offers **Declick Ramp** option that controls the declick ramp timing. Both **A**, **D**, **S** and **R** have **Range** options of: 10ms, 50ms, 100ms, 200ms, 500ms, 1000ms, 2000ms, 5000ms, 10000ms (default).
D	**Decay** (0-10000ms) Decay sets the transition time from 100% until reaching the *Sustain* level.
S	**Sustain** (dB) Sets the constant amplitude that is produced when a key is held. Note that when *Sustain* is set to maximum, the amplitude cannot drop during the *Decay* stage. The right click menu offers Expression option.
R	**Release** (0-10000ms) Sets the time for the sound to fade from *Sustain* level to 0 when key is released.
BIPOLAR	**Bipolar** When ON the module produces a signal in the amplitude range [–1;1]. When OFF it only uses the amplitude range [-1;0] (Unipolar).
CLK	**Click** Clicks are spikes in the envelope caused by quick changes in amplitude, e.g., when receiving many triggers in a sequence. When ON the module doesn't handle clicks in any special way, it just let's them pass (useful for percussion). When OFF the module ramps off clicks (default). See **Attack** knob for additional settings.
TRIGGER	**Trigger** When ON the module restarts the envelope on MIDI note-on events.
INV	**Invert** Inverts (or flips) the ADSR signal. Useful for say outputting in the [0;1] range in unipolar mode, or to simply generate an inverted ADSR envelope.
CURVE	**Curve** Selects the active envelope curve. Choose between **Linear**, **Exp** (Exponential), **Log** (Logarithmic) or **Hermite** (Smooth cubic curve): 

Envelope

Modulator

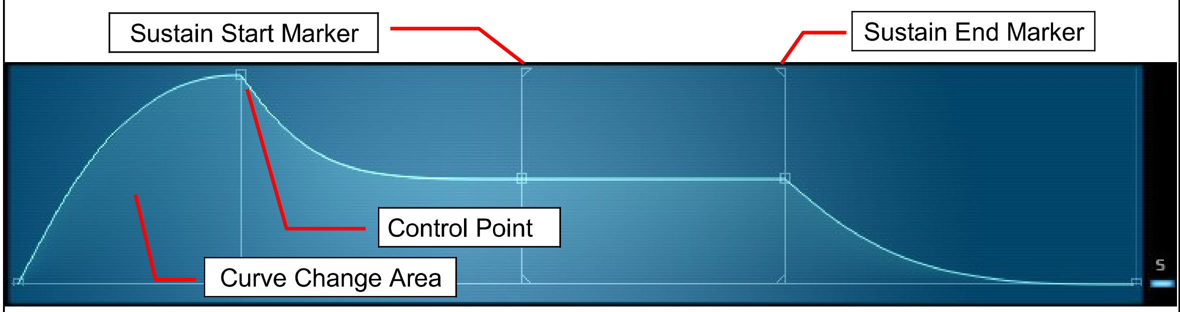


The Envelope module lets you draw a flexible envelope curve using simple control points. The curve segments between control points have adjustable slopes and 4 different curve types are supported: Linear, Exp, Cubic and Hermite. Moreover the overall duration as well as the overall slope of the envelope can be changed, and *sustain* can optionally be enabled to loop a part the curve while a key is held.

Next to the display are two panel buttons (top-most/right). The top panel button toggles an *Expanded* mode, where only the Envelope display is shown, but at twice the size. To the same effect, the panel button below can toggle the waveform display and all knobs & LEDs on/off when not in Expanded mode.

Parameters	
AMP	**Amplitude** (dB) Sets the maximum amplitude of the envelope.
DURATION	**Duration** (ms) Sets the total duration of the envelope. A **Range** option allows for tweaking the maximum duration range. The **Tempo-Sync** option allows the duration to sync to the host tempo.
SLOPE	**Slope** Offsets the slope settings globally for all curves in the envelope.
OFFSET	**Offset** Sets the start offset into the envelope, where playing will start on note trigger.
BIPOLAR	**Bipolar** When ON the module produces a signal in the amplitude range [–1;1]. When OFF it only uses the amplitude range [-1;0] (Unipolar).
TRIGGER	**Trigger** When ON the envelope is triggered/detriggered on MIDI note-on & off events.
CLK	**Click** Clicks are spikes in the envelope caused by quick changes in amplitude, e.g., when receiving many triggers in a sequence. OFF means clicks will be ramped (default). ON disables ramping, leaving any clicks as they were.
INV	**Invert** Inverts (or flips) the envelope. Useful for outputting in the [0;1] range in unipolar mode, or to simply generate an inverted envelope.
CURVE	**Curve Type** Choose between Linear, Exp(onential), Cubic or Hermite. All curves, except Linear, have adjustable slopes. Default curve is Exp.
S	**Sustain** When ON this enables looping of an envelope region marked by the sustain start and end markers, for as long as the currently played key is held.
PANEL BUTTONS	**Panel Buttons** The Top-most panel button switches Envelope to *Enlarged mode*. Second panel button switches Envelope to *Collapsed mode*, showing only the Envelope display and hiding all knobs/switches. Large mode is always collapsed.

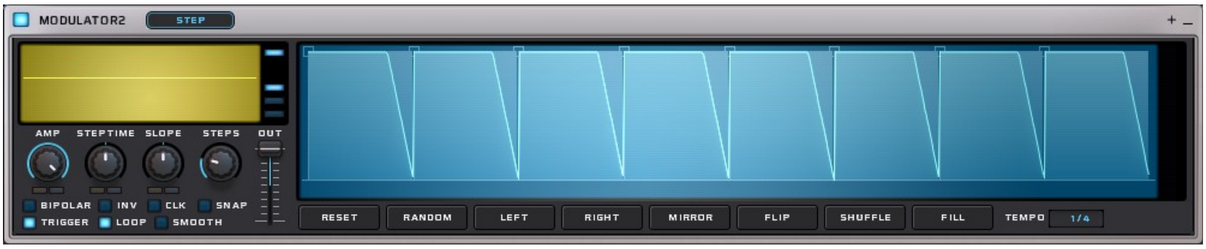
Envelope Display



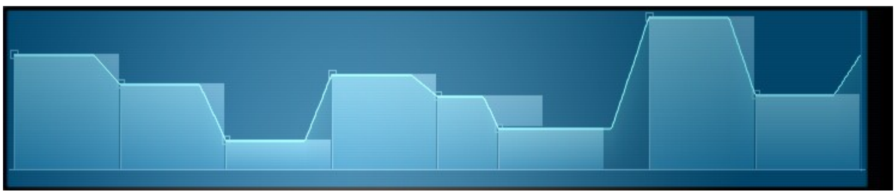
	Double Click on Display Area Adds a new Control Point.
	Left Click on Control Point + Drag Moves Control Point. A control point cannot be moved past other control points. Envelope control points will also snap to zero amplitude
	Left Click on Sustain Marker + Drag Moves Sustain Marker left or right. The markers snap to the control points.
	Left Click on Curve Area + Drag up/down Adjust slope for the selected curve segment.
	Right Click on Control Point Removes a Control Point.
	Double Right Click on Curve segment Flip slope. This is especially useful in Exponential mode, as it flips the direction of the non-symmetric Exponential slope.
	Mouse Wheel + Optional Drag Zoom and pan horizontally.
	Shift + Mouse Wheel + Optional Drag Zoom and pan vertically.
	Double Click Mouse Wheel Reset Zoom.

Step

Modulator



The Step module is a programmable step sequencer, with 1 to 32 steps. Each step's length and decay slope can be adjusted individually. Each step's start position can also be adjusted in order to have steps start or stop off-beat. The global step-time as well as the global slope can be tweaked for all steps simultaneously. Finally the module features a tempo control and a number of useful buttons for editing and shuffling the step-sequence.

Parameters	
AMP	**Amplitude** (dB) Sets the maximum amplitude for the step sequence.
STEPTIME	**Steptime** (ms) Sets the steptime offset globally for all steps in the step-sequence.
SLOPE	**Slope** Offsets the decay slope settings globally for all steps in the step-sequence.
STEPS	**Steps** Adjusts the total number of steps in the step-sequence, from 1 to 32.
BIPOLAR	**Bipolar** When ON the module produces a signal in the amplitude range [–1;1]. When OFF it only uses the amplitude range [-1;0] (Unipolar).
INV	**Invert** Inverts (or flips) the step output. Useful for outputting in the [0;1] range in unipolar mode, or to simply generate an inverted step-sequence.
CLK	**Click** Clicks are spikes caused by quick changes in amplitude. OFF means clicks will be ramped (default). ON disables ramping, leaving any clicks as they were.
SNAP	**Snap** If the step module is used to modulate a step based parameter (e.g. notepitch note and octave), enabling Snap will force the step output to snap to the nearest meaningful value for this parameter.
TRIGGER	**Trigger** When ON the step sequence is triggered on MIDI note-on events. When OFF the step sequencer runs continuously (free running).
LOOP	**Loop** When ON the step sequence will loop continuously. When OFF the step sequence will stop playing after a single run.
SMOOTH	**Smooth** When ON the step sequence will function more like a step based envelope, drawing a smooth curve instead of decaying steps: 

Step Display

	Left Click on Display Area Adjusts the individual amplitudes of the steps.
	Left Click on Control Point + Drag Moves Control Point (step start) to new amplitude and a potential off beat position.
	Shift + Left Click on Control Point + Drag Moves a step's Control Point (step start) without changing its amplitude.
	Double Click on Control Point Forces selected step to start on a beat position-beat.
	Mouse Wheel + Optional Drag Zoom and pan horizontally.
	Shift + Mouse Wheel + Optional Drag Zoom and pan vertically.
	Double Click Mouse Wheel Reset Zoom.
	Right Click Shows Step Preset Menu.

Step Buttons

RESET	**Reset** Resets all steps to their standard amplitude and position.
RANDOM	**Random** Sets random amplitudes for all steps.
LEFT	**Left** Shift step pattern left.
RIGHT	**Right** Shift step pattern right.
MIRROR	**Mirror** Flips step pattern horizontally, so the sequence runs backwards.
FLIP	**Flip** Flips amplitude levels (vertically).
SHUFFLE	**Shuffle** Mixes the current steps in a different order without affecting their amplitude.
FILL	**Fill** Copies all the used steps to all the unused steps as a repeated pattern. Useful when, e.g., going from 8 steps to say 16 steps.
TEMPO	**Tempo** Selects the active tempo with respect to the current song tempo (BPM).

HFO

Modulator

The HFO module is a *High Frequency Oscillator* that generates Saw, Square, Triangle, Ramp and Sine waveforms at frequencies up to 16kHz. You can think of it as the LFO's high-frequency brother, since it has the exact same knob parameters. For modulating parameters, a LFO is usually quite sufficient, but in some cases higher frequencies are needed. However, the HFO also has another trick up its sleeve: FM modulation. When in FM mode the HFO reads the current frequency control signal and produces a harmonic signal based on a FM Ratio control. For this reason the Rate knob is also replaced with a Ratio knob.

Parameters	
RATE	**Rate** (0.26Hz-16kHz) Sets the frequency of the oscillator.
RATIO	**Ratio** (only in FM mode) Sets the harmonic Modulator / Carrier ratio in FM mode. The Carrier frequency is read from the internal frequency control signal. Since the regular LFO module doesn't offer FM Ratio, a **Scale** option is available through the knobs right click menu, to downscale Ratio to reach lower frequencies.
AMOUNT	**Amount** (dB) Sets the maximum amplitude of the waveform, i.e., how "large" the waveform is.
FADE	**Fade** (0-100ms) Sets an optional fade-in time, in which the oscillator will slowly fade the amplitude from zero to max on trigger events.
PHASE	**Phase** Controls the phase of the oscillator, i.e., where in the waveform cycle are we ?.
BIPOLAR	**Bipolar** When ON the module produces a signal in the amplitude range [–1;1]. When OFF it only uses the amplitude range [-1;0] (Unipolar).
INV	**Invert** Inverts (or flips) the oscillator signal. Useful for outputting in the [0;1] range in Unipolar mode, or to simply generate inverted waveforms.
TRIGGER	**Trigger** When ON the module restarts the waveform on MIDI Note-on events. The waveform is restarted from the currently set phase value. Fade is also activated if Fade is set to anything above 0.
FM	**FM Mode** Enables FM modulation mode. In this mode the frequency control signal is used as source to produce a harmonically matching higher or lower frequency signal (controlled by Ratio) useful for FM modulating an oscillators frequency knob.
WAVE	**Waveform** Selects the desired oscillator waveform. Choose between: Saw, Square, Triangle, Ramp and Sine.

Decay

Modulator

Decay is a poor-mans envelope module. It only features a one-stage envelope, which starts at full amplitude and then decrease as defined by its decay time and decay waveform. However, unlike e.g. the ADSR, it also has a pre-delay setting and an optional loop time for repeating the decay envelope continuously.

Parameters	
DECAY	**Decay** (0-2000ms) Decay sets time for the amplitude to go from max to zero.
AMOUNT	**Amount** Sets the maximum amplitude of the waveform, i.e., how "large" the waveform is.
DELAY	**Delay** (0-2000ms) Sets the pre-delay before the decay envelope starts when triggered.
LOOP	**Loop** (0-2000ms) Sets the loop time, or how often the decay envelope should repeat. When Loop is set to 0, no looping is performed.
BIPOLAR	**Bipolar** When ON the module produces a signal in the amplitude range [–1;1]. When OFF it only uses the amplitude range [-1;0] (Unipolar).
INV	**Invert** Inverts (or flips) the decay envelope. Useful for outputting in the [0;1] range in unipolar mode, or to simply generate an inverted ADSR envelope.
TRIGGER	**Trigger** When ON the module restarts the decay envelope on MIDI note-on events.
SYNC	**Sync** Synchronizes the loop time to a whole number of beats.
WAVE	**Waveform** Chooses the Decay curve. Choose between Linear, Exp (Exponential) or Log (Logarithmic).

MidiTrig

Modulator

MidiTrig is a modulation module for receiving the common note and velocity MIDI triggers as a control signal.

Parameters	
NOTE	**Note Scale** Scales the incoming note value (0-127) by a common scale-factor.
VELOCITY	**Velocity Scale** Scales the incoming velocity value (0-127) by a common scale-factor.
MIX	**Mix Note/Velocity** Mixes the note and velocity signals. *Mix* sets the mix ratio. 127 means full *Velocity* signal. 0 means full *Note* signal.
AMOUNT	**Amount** Scales the resulting mixed signal.
BIPOLAR	**Bipolar** When ON the module produces a signal in the amplitude range [–1;1]. When OFF it only uses the amplitude range [-1;0] (Unipolar).
INV	**Invert** Inverts (or flips) the signal.
TRIGGER	**Trigger** When ON the module reads MIDI note/velocity data. When OFF the module does nothing.
RAMP	**Ramp** When ON the module attempts to reduce spikes in the signal by ramping them.

MidiData

Modulator

MidiData is a modulation module for receiving general MIDI data like Pitchbend, Aftertouch, and data from arbitrary MIDI controllers. You can also use the module in Manual mode to simply "draw" you own modulation using the value knob.

Parameters	
VALUE	**Value** Sets the value of the modulation signal in *Manual* mode. In any other mode, this knob has no effect.
CONTROL	**Control** Selects the MIDI controller id when in *Controller* mode. In any other mode, this knob has no effect.
BIAS	**Bias** Offsets the signal [-1:+1] in bipolar mode and [-0.5:+0.5] in unipolar mode. Default value is 0, no bias.
SMOOTH	**Smooth** Smooths the signal using a first order lowpass filter. 0 means no smoothing/lowpass filtering (default).
MIDILEARN	**Midi Learn** When enabled the module will wait for any MID controller to change and assign the *Control* knob to that controller. Typically, you turn on *MidiLearn*, change a MIDI controller on your MIDI device, and then turn *MidiLearn* back off again. You can now use you MIDI device as modulation source.
DELTA	**Delta** When OFF (default) controller movement is simply passed through. When ON controller movement is only passed through when a key is held (i.e., between note-on and note-off events) and only the *delta* relative to its value on note-on is outputted.
BIPOLAR	**Bipolar** When ON the module produces a signal in the amplitude range [–1;1]. When OFF it only uses the amplitude range [-1;0] (Unipolar).
INV	**Invert** Inverts (or flips) the signal.
TYPE	**Type** Choose between: *Pitchbend, Controller, Aftertouch, Timbre* and *Manual* mode.

EnvFollow

Modulator

EnvFollow is a simple envelope follower that reads the incoming signal and try's to construct an envelope that matches the amplitude level of the signal. It uses Attack and Release controls to tell how fast the envelope should rise and fall respectively. Unlike the previous modulation modules, EnvFollow therefore needs an incoming signal in order to work at all.

Parameters	
ATTACK	**Attack** (2-500ms) The attack time for the envelope follower, or how fast the envelope will rise.
RELEASE	**Release** (2-1000ms) The release time for the envelope follower, or how fast the envelope will fall.
SCALE	**Scale** Scales the signal by a common scale factor.
SMOOTH	**Smooth** Optionally smoothes the envelope (lowpass filter). 0 means no smoothing.
PEAK / RMS	**Peak / RMS** Selects whether the envelope follower should be based on Peak or RMS detection.
BIPOLAR	**Bipolar** When ON the module produces a signal in the amplitude range [–1;1]. When OFF it only uses the amplitude range [-1;0] (Unipolar).
INV	**Invert** Inverts (or flips) the signal.

S&H (Sample & Hold)

Modulator

S&H is a modulator for doing Sample and Hold – a well known modulation effect often used in electronic music. Its purpose is quite simple: Take an incoming signal and sample it at a given interval (rate) and output that value for the entire duration of the same interval. The module thus needs an incoming signal in order to work, typically a LFO. Built-in slew limiting (curve) controls the transition from one held value to the next.

Parameters	
RATE	**Sample Rate** (0-1kHz) The rate to use for sampling the incoming signal. High frequency means short interval, low frequency means long interval.
CURVE	**Curve** Adjust the slew rate transition curve from exponential to logarithmic. Center position (0) means soft linear slew transition, while –64 and +64 are near instant, exponential and logarithmic curves respectively.
PHASE	**Phase** Controls the phase of the sampling clock.
AMOUNT	**Amount** Sets how much of the S&H signal you wish to output by adjusting the mix ratio between the incoming- and the outgoing-signal. 127 means full S&H.
TRIGGER	**Trigger** When enabled, the sample clocks phase resets on note-on events.
SYNC	**Sync** Synchronizes the sampling rate to a whole number of beats. Useful for running modulation in sync to the song tempo.
SLEW	**Slew Direction** Select which direction to enable slew limiting. When going from low to high sample (up) or from high to low (down). Choose between Up, Down, Both or None (off). When slew limiting is off the output changes instantly (abrupt transition).

Shaper

Modulator
NEW IN V.2

Shaper takes an incoming signal and alters it by doing a simple lookup into a custom waveshaping function. The result is a wave-shaped modulation signal. The shaper-function is edited exactly like an envelope, using a set control points and curves to define the output signal. When using a linear ramp curve (default), the output is exactly the same as the input (disregarding any clipping at the ends of the curve due to overloaded input). Shaper is designed for low-frequency signals like LFO's, and does not perform any band-limiting. However, a smoothing parameter is available for lowpass filtering the resulting shaped output. The rightmost panel-switch enables a mode that shows the instantaneous lookup into the shaper-function.

Parameters	
SLOPE	**Slope** Offsets the slope settings globally for all curves in the waveshaping function.
BIAS	**Bias** Offsets the input [-1:+1] before look-up into the waveshaping function.
SMOOTH	**Smooth** Smooths the output using a first order lowpass filter. 0 means no smoothing/lowpass filtering.
AMOUNT	**Amount** Controls the dry/wet mix of the waveshaped signal with the input signal. 0 means 100% dry (input), while 127 means 100% wet (waveshaped).
BIPOLAR	**Bipolar** When ON (default) the input range of -1 to 1 is mapped to the lookup range. When OFF, the negative unipolar input range of -1 to 0 is mapped to the lookup range. This is useful for ADSR input for example.
CURVE	**Curve** Choose between **Linear**, **Exp(onential)**, **Cubic** or **Hermite**. All curves, except Linear, have adjustable slopes. Default curve is Cubic.

Control

Modulator
NEW IN V.2

Control is a modulator module that exposes the control signals that the synth uses internally. These signals travel backwards through the signal chain, starting at the output module and/or get altered or (re)created by any of the Controller modules that they pass through. Having the signals exposed in a separate modulator module makes it possible to route these signals into any parameter. The module can also be used for a quick inspection of control signals at a specific place in the signal chain, although the Scope module is recommended for this. The module allows for offsetting, smoothing, flipping and stretching the chosen control signal using the *Bias*, *Smooth*, *Polarity*, and *Scale* knobs. This should be sufficient for most use cases. If not, the signal can of course also be altered by routing it through other modules.

Parameters	
SCALE	**Scale** Scales the control signal from 0 to 2. Default is 1 (no scale).
POLARITY	**Polarity** Controls the (bipolar) amplitude polarity. Full right means positive (+1) and full left means negative (-1) polarity (inverted signal).
BIAS	**Bias** Introduces a DC Bias that offsets the signal (post scale and polarity) from -1 to +1.
SMOOTH	**Smooth** Filters the signal with a one pole lowpass smoothing filter. 0 means no filter.
BIPOLAR	**Bipolar** When ON the control signal is scaled by 2 and centered, normally resulting is a signal in the amplitude range [–1;1]. When OFF, the control signal is passed *as-is.*
SIGNAL	**Signal** Chooses between the 4 internal control signals: **Frequency** is the *frequency control signal* that is used by oscillators for tone generation and by filters for e.g. keyboard tracking. This is traditionally a positive unipolar signal. However, negative frequencies can make the output both bipolar and negative unipolar. **Phase** is the bipolar *phase control signal* that is used by oscillators for e.g. phase modulation. **Trigger** is the positive unipolar *trigger/gate control signal*, used for (re)triggering and gating in e.g. the Amplifier module. 1 means note-on and 0 note-off. **Note** is the positive unipolar *note control signal*, which essentially is equivalent to the incoming stream of MIDI notes. This is the signal that is usually converted into the frequency control signal using a Controller module.

Output

The Output module is in charge of sending the resulting audio signal to the host application (or in case of subpatches, to the parent patch). However, all incoming data from the host (MIDI events) also passes through the Output module. This means that modules only receive this data if they are connected, at some point in the signal chain, to the output module. Only one instance of the Output module is allowed and it is always there (it cannot be removed). The module has Volume, Panning, DC Removal and Clipping parameters for tweaking the output before it is sent to the host.

Parameters	
VOLUME	**Volume** Amplifies or Reduces the volume of the incoming signal. Centered means no change in volume. The Volume knob is always directly linked to the output knob on the Control Panel. The **Volume Range** [+6dB (default), +12dB or +24dB] can be set through the right-click menu. There is also a **Headroom** setting available, where you can choose the synth's output's desired headroom. Choose between the default 85% (-16.5dB), 50% (-12dB), 25% (-6dB) and 0% (0dB). When the synth is used as an instrument it is recommended that this is left at the default setting to avoid channel clipping in the host, especially for polyphonic patches. When the synth is used as an effect however, the Headroom defaults to 0%. In this case, the input module will not have to take the headroom into account, meaning that levels / thresholds etc. inside the synth, match the actual input signal. Headroom is always 0% and hence disabled in subpatches.
PAN	**Pan** Pans a stereo signal left or right.
DC	**DC Removal** (0Hz[off] – 60Hz) Optionally removes very low frequencies from the output signal.
CLIP	**Clip** Optionally clips the signal to a specified number of dB before outputting. A setting of 127 means no clipping (OFF).

Acknowledgments

This software was created by Kasper Nielsen, starting out in 1998 with a computer and a dream of creating a modular software synthesizer. Many years have gone by, and countless days and sleepless nights have been spent staring into the screen to make this software what it is today. Yet, this work would not have been possible without help and inspiration from these friends, individuals, music professionals and companies.

Special thanks to:

Emil Arffmann Hansen
Jan Marguc
Andreas and Stavros Kamenidis (Predators)
Olli Niemitalo
Juan Antonio Arguelles Rius (Arguru)
Linus Elman (Probe)
Nasca Octavian Paul
Niels Bye Nielsen (Bye Lye)
Bjørn Jacobsen (Cujo)
Mircea Marghidanu (Nervus)
Bill Halsey (Cosmosis)
Jeroen van Garling (Synchro)
Scot Solida
Dan Steele
Jan Larsen
Holger Lagerfelt
Richard Christy
Stefan Pattis
Oskari Tammelin (Jeskola)
Laurent de Soras
Michael Herf
Alf-Andre Walla (MadGonzo)
Laurent Gaudin (ksn)
Chris Menzi (Wavetraxx)
Geoff Heade (BluffMonkey)
Bryan Lee (Xenos Soundworks)
Robert Bristow-Johnson
Vadim Zavalishin
Lode Vandevenne
Paul Nettle
Will Pirkle
Udo Zölzer
Sophocles J. Orfanidis
Bob Moog
Computer Music Magazine
Voger Design
Steinberg for VST
Apple for AU
All KarmaFX Forum posters and bug reporters
All patch-bank creators
All skin makers and artists
All KarmaFX registered users, supporters and donaters!

Index