MPowerSynth

Overview

MPowerSynth is a deep synthesizer, featuring 3 oscillators, a noise generator, 2 filters and a modular effect pipeline. It also has 8 modulators, 8 multiparameters, an arpeggiator, a harmony generator, safety limiter and much more. The processing path is:

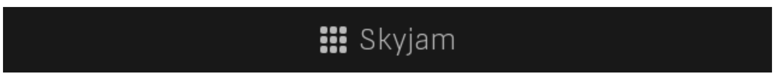
Voice1 = [Osc1] -> [Osc2] -> [Osc3] -> [Noise] -> [Filter1] -> [Filter2] -> Global ADSR ->
Voice2 = [Osc1] -> [Osc2] -> [Osc3] -> [Noise] -> [Filter1] -> [Filter2] -> Global ADSR -> + -> [FX]
...
"[xxx]" marks optional items that may be enabled or disabled.

Each of the 3 oscillators features our adjustable shape technology and provides the best sounding engine on the market, which doesn't suffer from "digital not analog" problems. Oscillator 1 is a little different from the others - it has a unison feature, which generates up to 10 voices with slight pitch differences and generates a full wide sound. Oscillators 2 and 3 do not provide unison, instead these offer different methods of merging the output from the previous section with the oscillator signal. By default the method is simply mixed, but you can also use frequency modulation, ring modulation, convolution and more. Both oscillators 2 and 3 can have a dedicated ADSR envelope.

Noise generator contains several noise options, plus a low-pass and high-pass filters. It can also have a dedicated ADSR envelope. It is mixed with the output of the previous section.

There are 2 deep filters available. Each one has a dedicated extended ADSR driving frequency, resonance and other parameters. Unlike classic filters the frequency is specified in octaves; that is a shift from the note frequency. So for example, if the note frequency is 100Hz and the frequency setting is +1 octave, then the central frequency of the filter is 200Hz, one octave higher than the note pitch.
There are more than 100 filter types available - classic LP/HP/BP/notch filters with slopes up to 120dB/octave, peak/shelf, harmonics, sub-X and other combined filters, format, comb/diffuser and polymorph filters. Most of them are pretty unique. The CPU consumption of different filters varies a lot; the most CPU-demanding ones are high-slope LP/HP/BP/notch filters, combined filters, diffusers and polymorph filters.

FX section presents the most advanced modular effect engine on the market, featuring more or less all our high-quality effects in a modular system. The engine is processing the output of all voices mixed together, so you don't need to worry about CPU that much anymore. Parameters of the effects loaded into the modular system cannot be automated, simply because you never know which effects that you will be using. However you can use modulators and multiparameters to control them, so if you need to automate any of them, you can just attach a multiparameter to that individual parameter and automate the multiparameter instead.



Presets

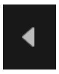
Presets button shows a window with all available presets. A preset can be loaded from the preset window by double-clicking on it, selecting via the buttons or by using your keyboard. You can also manage the directory structure, store new presets, replace existing ones etc. Presets are global, so a preset saved from one project, can easily be used in another. The arrow buttons next to the preset button can be used to switch between presets easily.

Holding **Ctrl** while pressing the button loads a random preset. There must be some presets for this feature to work of course.

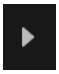
Presets can be backed up by 3 different methods:
A) Using "Backup" and "Restore" buttons in each preset window, which produces a single archive of all presets on the computer.
B) Using "Export/Import" buttons, which export a single folder of presets for one plugin.
C) By saving the actual preset files, which are found in the following directories (not recommended):
Windows: C:\Users\{username}\AppData\Roaming\MeldaProduction
Mac OS X: /Library/Application support/MeldaProduction

Files are named based on the name of the plugin like this: "{pluginname}.presets", so for example MAutopan.presets or MDynamics.presets. If the directory cannot be found on your computer for some reason, you can just search for the particular file.

Please note that prior to version 16 a different format was used and the naming was "{pluginname}presets.xml". *The plugin also supports an online preset exchange. If the computer is connected to the internet, the plugin connects to our server once a week, submits your presets and downloads new ones if available. This feature is manually maintained in order to remove generally unusable presets, so it may take some time before any submitted presets become available. This feature relies on each user so we strongly advise that any submitted presets be named and organised in the same way as the factory presets, otherwise they will be removed.*

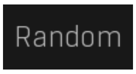


Left arrow
Left arrow button loads the previous preset.



Right arrow
Right arrow button loads the next preset.

Randomize
Randomize button loads a random preset.

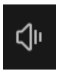


Randomize
Randomize button (with the text 'Random') generates random settings. Generally, randomization in plug-ins works by selecting random values for all parameters, but rarely achieves satisfactory results, as the more parameters that change the more likely one will cause an unwanted effect. Our plugins employ a smart randomization engine that learns which settings are suitable for randomization (using the existing presets) and so is much more likely to create successful changes.

In addition, there are some mouse modifiers that assist this process. The smart randomization engine is used by default if no modifier keys are held.

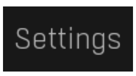
Holding **Ctrl** while clicking the button constrains the randomization engine so that parameters are only modified slightly rather than completely randomized. This is suitable to create small variations of existing interesting settings.

Holding **Alt** while clicking the button will force the engine to use full randomization, which sets random values for all reasonable automatable parameters. This can often result in "extreme" settings. Please note that some parameters cannot be randomized this way.



Sound
Sound button generates a note for the synthesizer to play. One click produces a note-on, the next one is a note-off. This can be useful to audition the current settings without using a MIDI keyboard or any other MIDI source to control the instrument.

Panic
Panic button resets the plugin state. You can use it to force the plugin to report latency to the host again and to avoid any audio problems. For example, some plugins, having a look-ahead feature, report the size of the look-ahead delay as latency, but it is inconvenient to do that every time the look-ahead changes as it usually causes the playback to stop. After you tweak the latency to the correct value, just click this button to sync the track in time with the others, minimizing phasing artifacts caused by the look-ahead delay mixing with undelayed audio signals in your host. It may also be necessary to restart playback in your host.
Another example is if some malfunctioning plugin generates extremely high values for the input of this plugin. A potential filter may start generating very high values as well and as a result the playback will stop. You can just click this button to reset the plugin and the playback will start again.



Settings
Settings button shows a menu with additional settings of the plugin. Here is a brief description of the separate items.

Licence manager lets you activate/deactivate the plugins and manage subscriptions. While you can simply drag & drop a licence file onto the plugin, in some cases there may be a faster way. For instance, you can enter your user account name and password and the plugin will do all the activating for you.

There are 4 groups of settings, each section has its own detailed help information: **GUI & Style** enables you to pick the GUI style for the plug-in and the main colours used for the background, the title bars of the windows and panels, the text and graphs area and the highlighting (used for enabled buttons, sliders, knobs etc).

Advanced settings configures several processing options for the plug-in.

Global system settings contains some settings for all MeldaProduction plugins. Once you change any of them, restart your DAW if needed, and it will affect all MeldaProduction plugins.

Dry/Wet affects determines, for Multiband plug-ins, which multiband parameters are affected by the Global dry/wet control.

Smart interpolation adjusts the interpolation algorithm used when changing parameter values; the higher the setting the higher the audio quality and the lower the chance of zippering noise, but more CPU will be used.

WWW

WWW button shows a menu with additional information about the plugin. You can check for updates, get easy access to support, MeldaProduction web page, video tutorials, Facebook/Twitter/YouTube channels and more.

Tab selector

Tab selector switches between subsections.

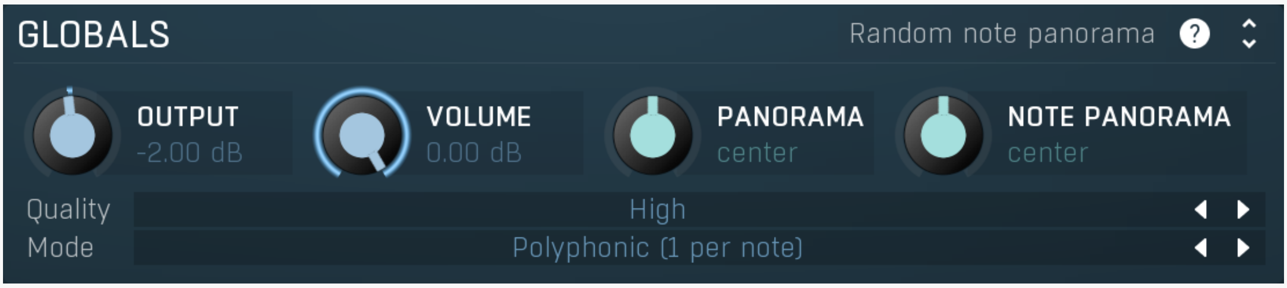
Randomize

Randomize button generates random settings for the tab.

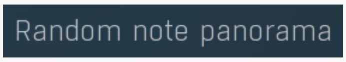
Presets

Presets button chooses a random preset for the tab.

Globals panel



Globals panel contains some basic global settings, such as output volume or panorama



Random note panorama

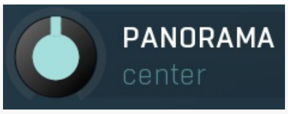
Random note panorama switch changes the behaviour of **Note panorama**. Normally that sets the panorama directly for each note, so you can for example use a modulator to control it. When this is enabled, it is merely a range and each note panorama will be randomly selected from the range -panorama..panorama. Hence it also doesn't matter if current panorama is right or left.

Output gain

Output gain defines the output gain. It is applied after all the generators and effects are performed.
Range: -48.00 dB to +48.00 dB, default 0.00 dB

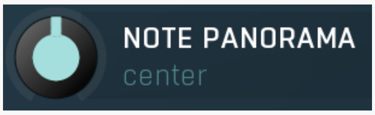
Volume

Volume defines the output volume adjustment, which is basically an alternative to gain with a different range. It is applied after all the generators and effects are performed.
Range: silence to 0.00 dB, default 0.00 dB



Panorama

Panorama defines the output panorama. It is applied after all the generators and effects are performed.
Range: 100% left to 100% right, default center



Note panorama

Note panorama defines the panorama-per-note. It is applied after all the generators and effects are performed, but unlike global **Panorama**, this may be different for each note. The value is assigned at the moment the note is pressed and isn't changed later.
Range: 100% left to 100% right, default center

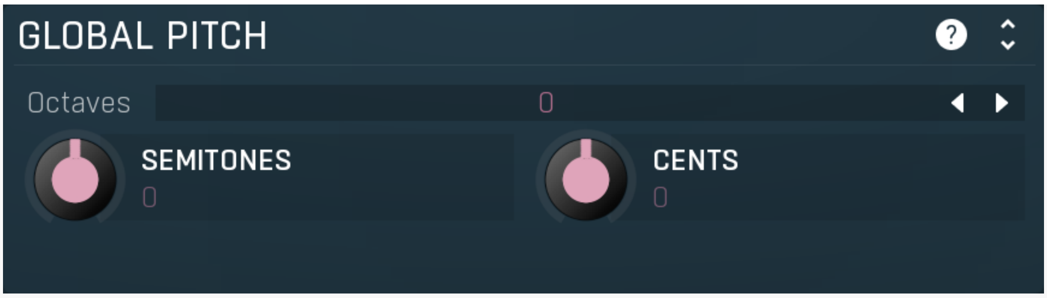
Quality

Quality controls the ratio between audio quality and CPU requirements used by the signal generator. In almost all instruments the signal generators are the weakest point, because generating digital signals is extremely CPU demanding if it shouldn't be prone to aliasing and digital distortion. Therefore almost all synthesizers on the market provide inferior quality, sometimes masking it by adding noise and claiming it to be analog simulations. Our generators on the other hand are the finest on the market and in most cases Medium mode will provide nearly perfect sound quality. If you are using saw-tooth waves, frequency modulation and other features creating extremely harmonically rich signals, you may consider using High or even Highest quality.
Low quality requires the least amount of CPU time, but you can experience aliasing (which may be used creatively of course). This is especially significant when using harmonically rich generator shapes, such as saw-tooth or square waves.
Medium activates an advanced generator, which requires slightly more CPU and memory resources, but minimizes aliasing. This mode is sufficient in most cases.
High improves the quality using more advanced interpolation.
Highest activates oversampling to at least 80kHz. This usually removes all aliasing and provides a true analog sound, in fact it provides a better than analog sound as it doesn't suffer from either analog or digital artifacts.
Extreme additionally uses extremely steep filters for the oversampling to minimize aliasing way below the limits of human hearing. This mode provides technically perfect generators. Its modifications such as **Extreme 16x** in addition increase the oversampling factor, which may be useful to remove distortion & aliasing when using transformations and additional mixing options such as frequency modulation. Please note that you can use the global oversampling as well, but this one is per-voice and may save CPU power as it is not oversampling the filters and FX section.

Mode

Mode defines the way in which the synthesizer works.
Polyphonic mode is the default mode, which makes the synth create as many voices simultaneously as necessary.
Polyphonic (single voice per note) is similar to polyphonic, but ensures that only one voice will be created for each MIDI key. If sustain is pressed (or other conditions cause multiple notes of the same key), the previous one will be terminated in the same way as when you release the key. This prevents voices stacking up when holding sustain pedal, however it may cause problems when harmonies are used, as pressing one key simulates pressing multiple keys.
Monophonic mode makes the synth play only one note at a time. When you hold one note and press another one the existing previous voice pitch is changed according to the new note, potentially with gliding. No attack stage or any kind of restart happens, so it sounds simply like the pitch has been changed.
Monophonic (toggle) mode is similar to the monophonic mode, however when you then release the new note while holding the previous one, the pitch of the voice is changed back to the note you are holding.
Monophonic (attack), **Monophonic (toggle + attack)** and **Monophonic (toggle + attack even off)** modes are similar to the 2 previous monophonic modes, however the voice is always switched back to the attack stage. This brings some focus to the new note, but no new voices are actually created. This mode smartly jumps in the ADSR envelope to the attack stage avoiding any abrupt changes in level and changes the pitch. The **even off** flag means that the attack is restarted even on note-offs. That means if you hold a long note, press a short one, then the attack is restarted not only when you press the short one, but also when you release it.
Monophonic (brutal) mode is similar to the previous modes, but the switch to the attack stage doesn't do any smoothing, so you may expect sharp clicking when the notes are restarting. This may be useful for short percussive sounds.
Monophonic (restart) mode and **Monophonic (toggle + restart)** mode are similar to the other monophonic modes, however instead of changing the pitch of the existing voices, the current voices are stopped in the same way as when you release them and new voices are created for the notes being pressed. This way the ADSR envelope is followed and you can get interesting overlaps of the release stages.
Trigger mode makes the synth completely ignore note release events and makes the notes be triggered on and off immediately providing a sort of staccato behaviour. This is especially useful for percussive sounds and in conjunction with another synth to form special attack sounds.

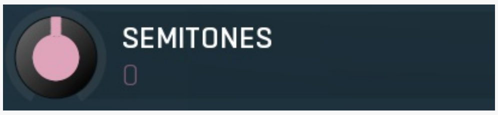
Global pitch panel



Global pitch panel lets you shift all generators and filters upwards or downwards.

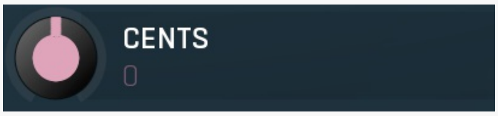
Octaves

Octaves defines the global pitch change in octaves.
Range: -8 to +8, default 0



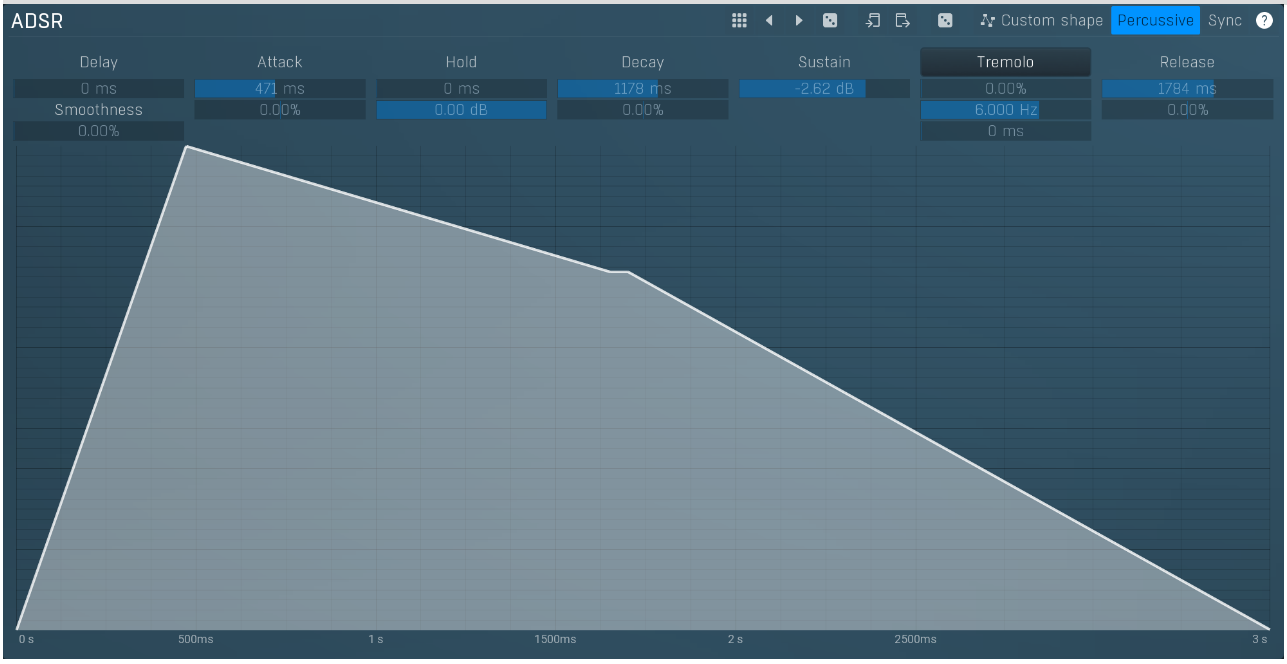
Semitones

Semitones defines the global pitch change in semitones.
Range: -24.00 to +24.00, default 0



Cents

Cents defines the global pitch change (in cents of a semitone). The actual pitch change is the sum of these 3 control values.
Range: -100.00 to +100.00, default 0



ADSR graph

ADSR graph controls the global ADSR envelope of the generated voices. Please note that additional oscillators, noise generators and filters can have their own ADSRs too.Our ADSR envelopes are much more sophisticated than classic attack-decay-sustain-release envelopes. Besides these common parameters they also let you control the curvature of each stage. Additionally, there are hold and delay sections ("DAHDSR"), global smoothing and tremolo. You can even use the custom shape mode to define your own attack/release curves.

Presets

Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.

Left arrow

Left arrow button loads the previous preset.

Right arrow

Right arrow button loads the next preset.

Randomize
Randomize button loads a random preset.

Copy
Copy button copies the settings onto the system clipboard.

Paste
Paste button loads the settings from the system clipboard.

Random
Random button generates random settings using the existing presets.

Custom shape
Custom shape button enables custom shape mode, which lets you draw your own attack and release stages using the envelope system. Both stages are then automatically connected to form the resulting envelope.

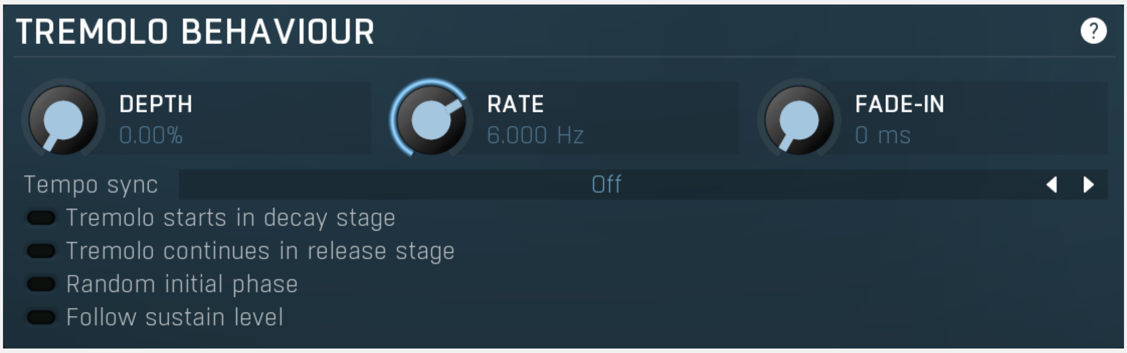
Percussive
Percussive button activates the immediate release mode in which case the note-off causes an immediate switch to the release stage. If this is disabled, the release stage does not occur until the whole attack/decay stage finishes.

Sync
Sync button controls the ADSR tempo sync feature. By default this is disabled and means that all times are followed exactly, meaning that if **Attack** is say 100ms, then it will be 100ms indeed. Tempo sync lets the plugin adjust the times to ensure it will be always in sync with the host tempo. In this case 100ms may become say 125ms if the tempo is 120bpm, because 125ms is the length of a 16th note. This makes it extremely simple to convert any envelope to a tempo-synced one. The plugin always chooses the nearest longer note, in other words it always round up.
Straight and **Triplets** modes automatically find 'nice' values.
For example, if a 16th note takes 100ms, the attack time is 550ms, and the sync mode is straight, then the plugin checks for 100ms, find out that it is too low, so it checks 8th note, being 200ms, still too low, then continues with quarter note, which takes 400ms, and still not enough, finally 800ms corresponding to a half note is the one, so the resulting time will be 800ms. Triplet cases are more complex, but the principle is the same.
1/16, **1/8** and **1/4** modes choose the nearest higher multiply of the base note length. For example, if a 16th note takes 100ms, the attack time is 550ms, and the sync mode is 1/16, the resulting time will be 600ms.

Tremolo
Tremolo button displays additional tremolo settings, containing tremolo behaviour and shape.

Tremolo settings

Tremolo behaviour



Depth

Depth controls the amount of tremolo mixed in the sustain stage (or potentially before).

Rate

Rate controls the tremolo rate and is relevant only if tempo sync is not used.

Fade-in

Fade-in controls the length of the tremolo fade-in. It is especially useful when you want to use the random initial phase feature to avoid the initial discontinuity when the tremolo kicks in.

Tempo

sync
Tempo sync lets you synchronize the tremolo to the host's tempo.

Tremolo starts in decay stage
Tremolo starts in decay stage makes the tremolo start during the decay stage. By default this is disabled and the tremolo starts in the sustain stage. When it is enabled you will most likely have a longer decay and also a longer tremolo fade-in, so that the tremolo slowly comes in as the envelope is decaying.

Tremolo continues in release stage
Tremolo continues in release stage makes the tremolo continue with the tremolo during the release stage. By default this is disabled and the tremolo stops as soon as the release stage starts.

Random initial phase
Random initial phase makes the tremolo start with a random phase. By default this is disabled and the tremolo starts always starts in the 0 phase, which ensures the tremolo always starts in the same way. However if you play multiple notes at once, the tremolo will be exactly the same, while you may want it to be different for each note and make it sound more 'human'. Enabling this option also activates a short **tremolo fade-in** to avoid initial discontinuity.

Follow sustain level
Follow sustain level makes the tremolo level based on sustain level. When this is disabled, the tremolo rarely reaches up to 100% level. However if the sustain level is say -20dB, then the tremolo actually cannot exceed 1% (which is -20dB), so it is clipped. It can however go upwards to 100%. This naturally changes the actual tremolo shape. If you want to avoid that and make sine really be a sine for example, enable this option, and in the case above the tremolo will really go up/down -20dB if set to 100%.

Presets
Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.

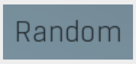
Left arrow
Left arrow button loads the previous preset.

Right arrow
Right arrow button loads the next preset.

Randomize
Randomize button loads a random preset.

Copy
Copy button copies the settings onto the system clipboard.

Paste
Paste button loads the settings from the system clipboard.



Random
Random button generates random settings using the existing presets.

Normal
Normal button switches the generator into the normal mode, which lets you edit the shape of the oscillator. This is especially advantageous for low-frequency oscillators, where the shape matters even though it doesn't have any physical meaning.

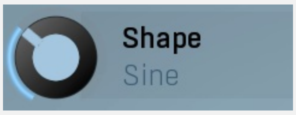
Convert
Convert button converts the current shape into harmonic-based representation. Please note that since the number of harmonics is limited, the result will not perfectly resemble the original shape.

Harmonics

Harmonics button switches the generator into the harmonics mode, which lets you edit the levels and phases of individual harmonics. This is especially advantageous for high-frequency oscillators, hence sound generators.

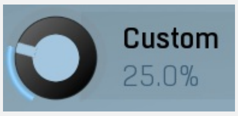
Signal generator in Normal mode

Signal generator in Normal mode works by generating the oscillator shape using a combination of several curves - a predefined set of standard curves, custom shape, step sequencer and custom sample. It also post-processes the shape using several filters including smoothing to custom transformations. This is especially useful when using the oscillator as an LFO (low-frequency-oscillator), where the harmonic contents does not really matter, but the shape does.



Shape

Shape controls the main shape used by the signal generator. There are several predefined shapes: exponential, triangle, sine power 8, sine power 4, sine square, sine, harmonics, more harmonics, disharmonics, sine square root, sine 4 root, rectangle, rect-saw, saw, noise and mess. You can choose any of them or interpolate between any 2 adjacent shapes using this control.



Custom

Custom controls the amount of the custom shape that is blended into the main shape.

Edit

Edit button shows the custom shape editor.

Step

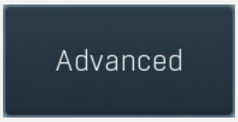
Step controls the amount of the step sequencer shape that is blended into the main shape (which has already been blended with the custom shape).

Edit

Edit button shows the step sequencer editor.

Smooth

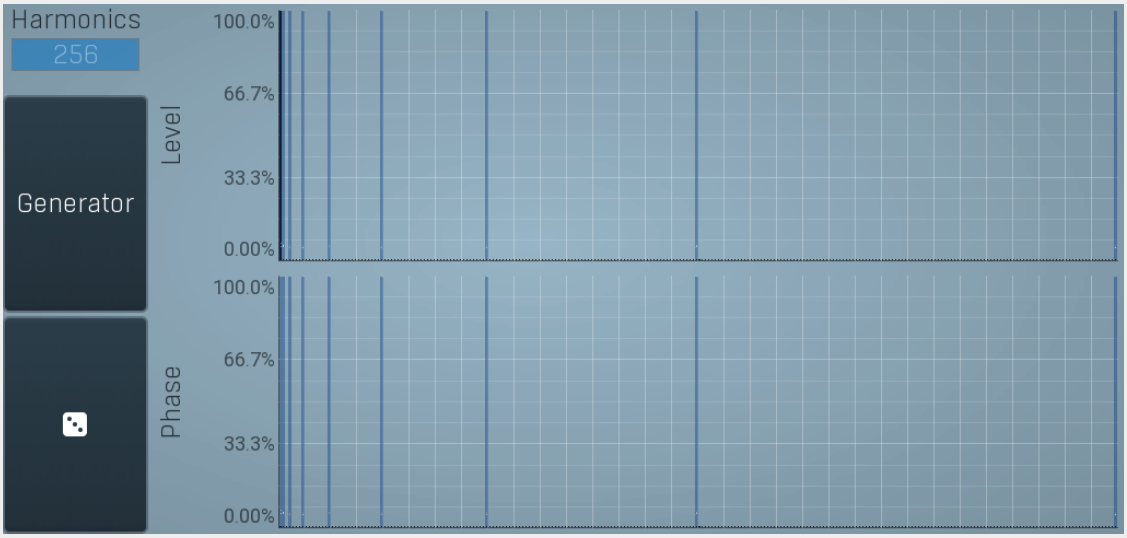
Smooth controls the amount of smoothing. Many shapes, especially those produced by the step sequencer, have rough jagged edges, which may be advantageous, but when used to modulate certain parameters, the output may be clicking or causing other artifacts. Smoothness helps it by smoothing the whole signal shape out and removing these rough edges.



Advanced

Advanced button displays an additional window with more advanced settings for post-processing the signal shape, such as harmonics or custom transformations.

Signal generator in Harmonics mode



Signal generator in Harmonics mode works by generating the oscillator shape using individual harmonics. Essentially a harmonic is a sine wave. The first harmonic, known as the fundamental, fits once in the oscillator time period, hence it is the same as selecting sine wave in the **Normal mode**. The second harmonic fits twice, the third three times etc. In theory, any shape you create in normal mode can be converted into harmonics. However, this approach to signal generation needs an enormous number of harmonics, which is both inefficient to calculate and mostly hard to edit. Therefore, the harmonic mode can process up to 256 harmonics, which is enough for very complex spectrums, however it is still not enough to generate an accurate square wave for example. If your goal is to create basic shapes, it is better to use the normal mode.

It is nearly impossible to say how a particular curve will sound when used as a high-frequency oscillator in a synthesizer, just by looking at its shape. Harmonics mode, on the other hand, is directly related to human hearing and makes this process very simple. In general, the more harmonics you add, the richer the sound will be. The higher the harmonic, the higher the tone. Usually, one leaves the first harmonic enabled too, as this is the fundamental tone, however you may experiment with more dissonant sounds without it.

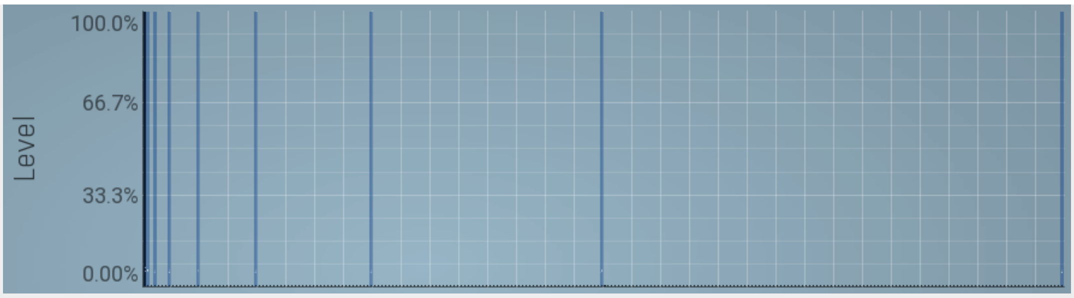
Editing harmonics can be time consuming unless you hear what you want, so a signal generator is also available. This great tool lets you generate a random spectrum by a single click. You can also open the **Generator** settings and edit its parameters, which basically control the audio properties in a more natural way - using parameters such as complexity, harmonicity etc.

Generator

Generator button shows a powerful harmonics generator, which can create unlimited number of various timbres and even analyze a sample and extract harmonics from it.

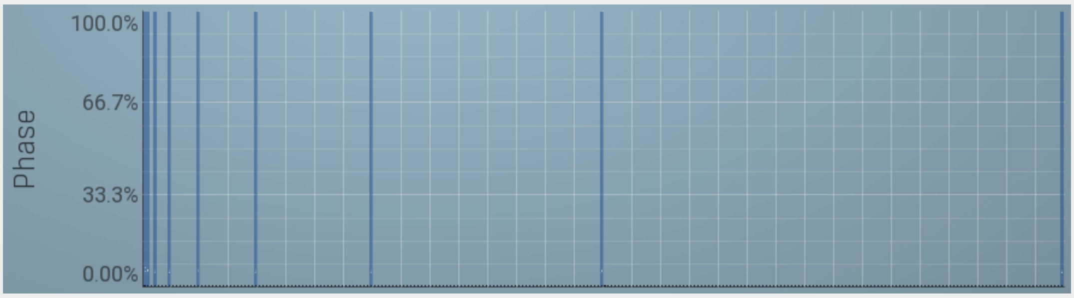
Randomize

Randomize button selects random parameters for the harmonics generator, so you can use it to get a random sound character instantly. Hold **Ctrl** to slightly modify existing generator settings instead of completely changing them.



Magnitudes graph

Magnitudes graph contains the levels of the individual harmonics. The highlighted bars are octaves, thus the 1st, 2nd, 4th, 8th harmonic etc.



Phases graph

Phases graph contains the phases of the individual harmonics. The highlighted bars are octaves, thus the 1st, 2nd, 4th, 8th harmonic etc.

0 ms **Delay**

Delay lets you shift the entire envelope forwards in time. While this doesn't make much sense for a global instrument envelope for instance, it may be well useful to control characteristics of evolving sounds.

471 ms **Attack**

Attack controls the length of the initial stage of the envelope. It is one of the most important parameters controlling how quick the initial transient is. For most instruments the length is quick short, but for pads and other slowly evolving sounds it is quite common to set this to several seconds.

0 ms **Hold**

Hold specifies the time the level stays at maximum after the attack stage.

1178 ms **Decay**

Decay controls the time it takes for the level to drop from the maximum to the **Sustain**. If the sustain is 0dB, then this parameter has no effect, because in a way the sustain stage starts immediately after the attack.

-2.62 dB **Sustain**

Sustain controls the sustain level. For most sounds the initial attack transient is the highest point of the entire sound. Imagine playing a string instrument, such as a guitar, the initial hit to the strings is represented by the attack+hold+decay sections and is the most prominent. After that the level drops to the sustain stage, where it holds for most of the time.

0.00% **Tremolo**

Tremolo defines the amount of the tremolo effect that is engaged in the sustain, or even in the decay section and continues until the envelope ends. While this is a rather unusual feature for an envelope to have, it is very handy for simulating various effects human players do when performing on real instruments, such as the tremolo or vibrato.

1784 ms **Release**

Release controls the length of the release section, which usually starts when a note is released.

0.00% **Attack shape**

Attack shape controls the shape of the attack section and defines its sound character.

0.00 dB **Hold level**

Hold level controls the level of the hold section. By default it equals maximum meaning that the hold section actually holds the maximum level. However by making it lower you can sort of simulate 2 separate decay sections, first going from maximum to hold level, second going from hold level to sustain.

0.00% **Decay shape**

Decay shape controls the shape of the decay section and defines its sound character.

6.000 Hz **Tremolo rate**

Tremolo rate controls the speed of the tremolo. In the tremolo settings it is possible to control additional characteristics including tempo sync.

Release shape

Release shape controls the shape of the release section and defines its sound character.

Smoothing

Smoothing lets you smoothen the entire envelope avoiding abrupt jumps. Note that in some cases involving short jumps the results may be a bit obscure.

Tremolo fade-in

Tremolo fade-in defines the time for the tremolo to reach its full level. It is a natural behaviour of human players (on say a saxophone) that they don't start a full tremolo immediately and rather let the modulation rise to maximum over a period of time.

Osc 1 panel

Osc 1 panel contains settings for the main oscillator. This first oscillator is different from the other oscillators as it doesn't offer any combination mode, but it has unison feature instead. In most cases you will leave this oscillator enabled. For noise-based settings you may want to disable it though.

Presets

Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.

Left arrow

Left arrow button loads the previous preset.

Right arrow

Right arrow button loads the next preset.

Randomize

Randomize button loads a random preset.

Copy

Copy button copies the settings onto the system clipboard.

Paste

Paste button loads the settings from the system clipboard.

Random

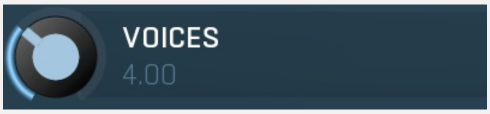
Random button generates random settings using the existing presets.

Unison panel

Unison panel contains parameters of the unison generator, which essentially creates multiple clones for each voice making the sound rich and wide.

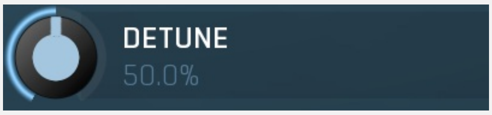
High detune

High detune extends the detuning range significantly making it an interesting tool for creative effects.



Voices

Voices controls the number of voices of the unison generator. The number may be fractional in which case the last voice is not "fully generated".
Range: 1.00 to 10.00, default 4.00



Detune

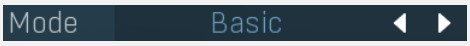
Detune defines the amount of detuning for each unison clone.
Range: 0.00% to 100.0%, default 50.0%

Shape

Shape controls the shape of the frequencies for each unison voice. At 0% the frequencies will be equally displaced around the base pitch. At +100% most of the frequencies will be placed closely around the base pitch. Conversely, at -100%, they will tend towards the extremes and be as far away from the base pitch as possible, the limit being defined by the **Detune** parameter.
Range: -100.0% to 100.0%, default 0.00%

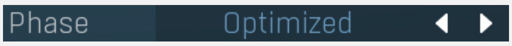
Widening

Widening controls the stereo widening for the unison clones.
Range: Mono to 200.0%, default 0.00%



Mode

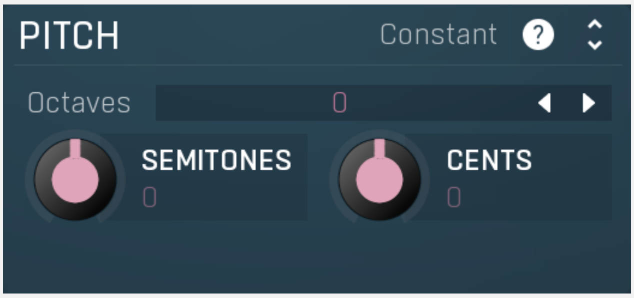
Mode controls the way the unison voices cover the frequencies around the base pitch, hence the actual unison sound.
Basic is the original mode in which the voices are surrounding the base pitch, but the actual pitch may not actually be created unless the number of voices is at the maximum.
Advanced covers the frequencies around the base pitch up to the detune level, and the dispersion is controlled by the **Shape** parameter.
No base pitch is similar to Advanced mode, but similarly to the Basic mode it does not generate the actual base pitch.



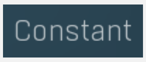
Phase

Phase controls the initial phase for the unison voices. The voices share the same signal shape and are very similar in frequency, so they interact quite a bit when their phases are getting close to each other. This causes natural phasing. This mode controls at which phase the voices start, which directly defines the character of the phasing, mainly in the beginning of the note.
Optimized is the default value, which optimizes the phasing to the minimum in most cases.
Optimized wide is similar but provides the maximum stereo width. Please note that this may cause mono-compatibility problems.
Random provides random phases every time. This will result in a slightly different character for each note.
Random wide is similar but provides the maximum stereo width. Please note that this may cause mono-compatibility problems.
Zero lets all voices start at the initial phase, therefore you may expect maximum phasing to occur at the beginning and slowly to disappear. Please note that this mode may cause the unison effect to disappear for minimum detune.

Pitch panel



Pitch panel lets you shift the main generator upwards or downwards.

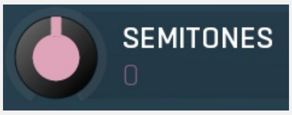


Constant frequency

Constant frequency makes the main oscillator ignore the note pitch and behave as if an A4 (440Hz) note is received each time.

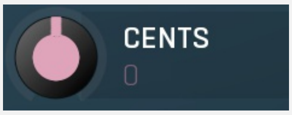
Octaves

Octaves defines the main oscillator pitch change (in octaves).
Range: -8 to +8, default 0



Semitones

Semitones defines the main oscillator pitch change (in semitones).
Range: -24.00 to +24.00, default 0



Cents

Cents defines the main oscillator pitch change (in cents of a semitone). The actual pitch change is the sum of these 3 control values.
Range: -100.00 to +100.00, default 0

Transform panel

Transform panel contains some additional oscillator transformation features.

Mode

Mode controls the type of transformation used on the oscillator waveform. The oscillators themselves are pre-processed in a very complex manner to provide maximum audio quality. However that means that every time you change something in the oscillator editor, lots of calculations need to be performed, which may take quite some CPU. The transformations however are performed without pre-processing, so they may provide lower audio quality, but they are much less CPU intensive and often provide actually higher quality when modulated, because these are always sample accurate.

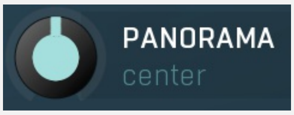
It is therefore recommended to use the oscillator editor for static sounds and then use transformations for modulation. Another reason to use the transformation is the actual harmonic properties of the generated sound - any oscillator (algorithmic or wavetable) produces only a fundamental frequency and its harmonics, the actual wave shape only controls the levels and phases of these frequencies. This means that if the fundamental pitch is say 100Hz, then the generated sound contains only 100Hz, 200Hz, 300Hz etc. It can never produce 150Hz for example. And that's when the transformations comes in handy again. When you transform the waveform in any way the result is just another waveform, so it still cannot produce other frequencies. However if you change the transformation depth in real-time, say using modulators, the output could produce just about any frequencies, all depending on the actual transformation and the way it is modulated.

Disabled disables any transformation.

PWM implements several pulse-width modulation algorithms, all affecting the width of the pulse or bending its shape in time. PWM tends to produce frequencies around each harmonic when modulated. The quicker the PWM depth is changed, the further these frequencies are from the main harmonics and the sound gets richer until a point where it is completely disharmonic.
Sync essentially duplicates the periods of the waveform (not necessarily a multiple). The higher the depth, the higher the dominant harmonic, so moving the depth in a way controls the pitch without losing the sense of the actual fundamental pitch. Sync waveform has abrupt edges in most cases. These generate lots of higher harmonic content, sometimes even reaching zipper noise. To diminish this and to get a cleaner sound, use the windowed versions, which smooth out the edges.
ASync transformations are similar to Sync, however the waveforms are bent, so the typical sync screaming is less resonant and more inharmonic.
Bending and mirroring alter the waveform quite smoothly, so these produce less rich tones and the effects may remind you of phasing. Discontinuities may occur when modulated quickly, so you can exploit that to get a richer sound if needed.
Reversing merges the waveform with a reversed replica of itself. This may often cause abrupt edges producing a sort of a modulated square wave. To get the most out of it, the original waveform should not be (anti)symmetric - when you look at a sine wave for example, the right half is just an inverted left half, so reversing it isn't such a miraculous action. This is true for most basic waves including saw and rectangle for example. Therefore reversing is most interesting when used with some more complex wave-shapes produced by harmonic mode or the step sequencer etc.
Invert, Maximize, Zero under, Power and Quantize are manipulating the amplitude only and due to the discontinuities these produce fairly rich sounds, lo-fi in a way. Please note the processor may produce different results for different quality modes, because it can generate specific waveforms for different notes to improve quality, but since the amplitude will be different, the results of the transformation will be different as well.
AM provides amplitude modulation with itself or a reversed version of itself. As such it provides minimum non-harmonic frequencies.
Recursive transformations are the most complex and basically transform the waveform by projecting it onto itself. The output generally depends on the input waveform signal and you might say that the more complex the initial waveform is, the more complex the output waveform will be. Recursive transformations may produce extremely full spectra and may mix the properties and richness of other transformations combined. As such, simple waveforms such as sine and similar usually produce the best results. Modulating a harmonically complex waveform can easily end up with white noise.

Depth

Depth controls the amount/type of the selected transformation and could be used for modulation. Its meaning depends on the selected transformation.
Range: 0.00% to 100.0%, default 50.0%



Panorama

Panorama defines the oscillator panorama transformation.
Range: 100% left to 100% right, default center

Shape graph editor

Shape graph editor controls the oscillator shape. You can use either oscillator shapes or harmonics; the latter are preferred in this case as they directly control the harmonic structure of the generated sounds. Note that displaying the output shape using an oscilloscope may produce very different images than what you see here. That is not a bug. It's the plugin ensuring it sounds as good as possible, analogue-like.Signal-generator is an incredibly versatile generator of low & high frequency signals. It offers 2 distinct modes - Normal and Harmonics.

Normal mode is appropriate for low-frequency oscillators, where the graphical shape is relevant and is used to drive some form of modulation. For example, a tremolo uses this modulation to change the actual signal level in time. Frequencies for such oscillators usually do not exceed 20Hz as this is a sort of limit above which the frequencies become audible.
Harmonics mode is designed for high-frequency oscillators, where the actual shape is not as important as the harmonic content of the resulting signal, hence it is especially useful for actual audio signals. Please note that since a shape can contain more harmonics than those available from the harmonic generator, the results may not be exactly the same. As an example, a rectangular wave in normal mode may sound fuller than when converted to the harmonic mode.

Use the arrow-down button to switch from normal mode to harmonics mode or click the **Normal** and **Harmonics** buttons

Normal mode

The generator first uses a set of predefined signal shapes (sine, triangle, rectangle...), which you can select directly by right-clicking on the editor and choosing the requested shape from the menu. This menu also provides a link to the modulator shapes preset manager, normalization and randomization. You can also use the **Main shape** parameter, which generates a combination of adjacent signals to provide a nearly inexhaustible number of basic shapes.

The engine then combines the predefined shape with a **Custom shape**, which may be anything you can draw using the advanced envelope engine, depending on the level set by the **Custom shape** control. Use the **Edit** button to edit the custom shape.

You can also combine those results with a fully featured step sequencer, with variable number of steps and several shapes for each of them, depending on the level set by the **Step sequencer** control. Use the lower **Edit** button to edit the step sequence.

Those results may be mixed with a custom sample, which is available from the advanced settings, accessed by clicking the **Advanced** button.

Smoothness softens any abrupt edges, generated by the step sequencer for example.

Finally there are **Advanced** features providing more complex transformations, adding harmonics etc. or you can click the **Randomize** button in the top-left corner to generate a random, but reasonable, modulator shape.

Harmonics mode

Harmonics mode represents the signal as a series of harmonics (that is, multiples of the base frequency). For example, when your oscillator has a frequency of 2Hz (set in the **Rate** panel), then the harmonics are 2Hz, 4Hz, 6Hz, 8Hz etc. In theory, any signal can be created by mixing a potentially infinite number of these harmonics.

The harmonics mode lets you control the levels and phases of each harmonic. The top graph controls the levels of individual harmonics, while the bottom one controls their phases. Use the left-mouse button to change the values in each graph, the right-mouse button sets the default for the harmonics - 0% level and 0% phase. In both graphs the harmonics of power 2 (that is octaves) are highlighted. Other harmonics may actually sound disharmonic, despite their names.

For example, if you reset all harmonics to the defaults and increase only the first one, you will get a simple sine wave. By adding further harmonics you make the output signal more complex.

Harmonics controls the number of generated harmonics. The higher the number is, the richer the output signal is (unless the levels are 0% of course). This is useful to make the sound cleaner. For example, if you transform a saw-tooth wave to harmonics, it would not sound like a typical saw-tooth wave anymore, but more like a low-passed version of one. The more harmonics you use, the closer you get to the original saw-tooth wave.

Generator is a powerful tool for generating the harmonics, which are otherwise rather clumsy to edit. The generator provides several parameters based upon which it creates the entire series of harmonic levels and phases. These parameters are usually easier to understand than the harmonics themselves. Part of the generator is the randomizer available via the **Random seed** button, which smartly generates random settings for the generator. This makes the process of getting new sounds as simple as possible.

Signal generation fundamentals

The signal generator produces a periodic signal with specified wave shape. This means that the signal is repeating over and over again. As a result it can only contain multiples of the fundamental frequency. For example, if the generator is producing 100Hz signal, then it can contain 100Hz (fundamental or 1st harmonic), 200Hz (2nd harmonic), 300Hz (3rd harmonic), 400Hz (4th harmonic) etc. However, it can never produce 110Hz. You can then control the level of each harmonic and their relative phases. It does not matter whether you use the normal mode using oscillator shapes, or harmonics mode where you can control the harmonics directly. If both modes result in the same wave shape (such as sine wave vs. 1st harmonic only), then the result is exactly the same.

Sine wave is the simplest of all as it contains the fundamental frequency only. The "sharper" the signal shape is, the more harmonics it contains. The biggest source of higher harmonics is a "discontinuity", which you can see in both rectangle and saw waves. In theory, these signals have an infinite number of harmonics. However since our hearing is highly limited to less than 20kHz, the number of harmonics which are relevant is actually pretty small. If you generate a 50Hz signal, which is very low, and assuming that you have extremely good ears and you actually hear 20kHz, then the number of harmonics audible for you is 20000 / 50 = 400.

What happens above 20kHz?

Consider the example above again, what happens with harmonics above 400? These either stay there and simply are not audible, disappear if anti-aliasing is used, or get aliased back under 20kHz in which case you get the typical digital dirt.

When you convert a rectangle wave to harmonics mode, only the first 256 harmonics are used, so it basically works like an infinitely steep low-pass filter. What is the limit then? 50 Hz * 256 = 12.8kHz. The harmonic mode will not produce anything above this limit if you are generating a 50Hz signal. Most people do not hear anything above 15kHz, so this is usually enough, but if not, you may need to use the normal mode where you get the "infinite" number of harmonics.

What you see is not always what you get!

Say you want a rectangle wave and play a 440Hz tone(A4). You would expect the output signal to be a really quick rectangle wave, right? Wrong! If you would do that, and actually most synthesizers on the market do that, you would get the infinite number of harmonics. And, since you are working in say 48kHz sampling rate, the maximum frequency that can actually exist in your signal is 24kHz. So everything above it would get aliased below 24kHz, and there would be a lot of aliased dirt.

The "good" synthesizers perform a so-called anti-aliasing. There are several methods, most of them require quite a lot of CPU or have other limitations. The goal is to remove all frequencies above the 24kHz in our case or in reality, it is more about removing all aliased frequencies above 20kHz - this means, that we do not care about frequencies above 20kHz, because we do not hear them anyway. But we will keep it simple. Let's say we remove everything above 20kHz. You already know that the rectangle wave can be created using an infinite number of harmonics or sine waves. We removed everything above the 45th harmonic (20000 / 440) so our rectangle wave is trying to be formed using just 45 harmonics, so it will not really look like a rectangle wave.

After some additional filtering (like DC removal), the rectangle wave may look completely different than a true rectangle wave, yet it would sound the same! Does it matter? Not really. You simply edit the shape as a rectangle wave and let the synthesizer do the ugly stuff for you. But do not check the output, because it may be very different than what you would expect ;).

How can I generate non-harmonic frequencies?

Ok, so now you are playing a 440Hz (A4) saw wave, it contains 440Hz, 880Hz, 1320Hz etc. Anything generated using the signal generator can contain only these frequencies, the only difference is the levels and phases of each of them. What if you want to make the signal dirty by adding say 500Hz? Well, that is not that simple! Here we are getting into audio synthesizer stuff, so let us just give you a few hints.

The traditional way is to use modulation. One particular method is called frequency modulation (FM). Instead of generating a 440Hz saw wave with your generator, you change the pitch, up and down. You are modulating the frequency, that's why FM. It is basically a vibrato, but as you increase the speed of the vibrato, it gets so quick that you stop noticing the pitch changes (that's very simplified but it serves the purpose) and instead it starts producing a very complex spectrum. Will the 500Hz be there? Well, if setup correctly, yes, but there will also be lots of other non-harmonic frequencies.

Another way is possible without any other tools. Let's say you do not want 440Hz, but 660Hz. Then you may generate 220Hz instead of 440Hz (which is one octave below it) and voila, 660Hz is the 3rd harmonic (3 x 220 is 660)! But you need to shift the saw wave one octave above. Fortunately it is not that hard here - go to the normal mode, select saw tooth, click advanced, and use the harmonics panel to remove the fundamental and leave just the 2nd harmonic, then convert it to harmonic mode. Well, it's not that hard, but it's not exactly simple either...

The only way is, of course, additive synthesis. In that case you do not use one oscillator, but many of them. It lets you generate just about anything. But there is a catch, actually many of them. First, you need to say "ok I want this frequency and that frequency...", the setup is actually infinitely hard as there may be an infinite number of frequencies :). And the second is, of course, CPU requirements.

So is there some ultimate solution? Nope, sorry. The good thing is, you will not probably need it, because while what you see is not always what you get, also what you want is often not what you really want to hear :).

Presets

Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.

Left arrow

Left arrow button loads the previous preset.

Right arrow

Right arrow button loads the next preset.

Randomize

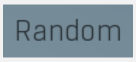
Randomize button loads a random preset.

Copy

Copy button copies the settings onto the system clipboard.

Paste

Paste button loads the settings from the system clipboard.



Random

Random button generates random settings using the existing presets.

Normal

Normal button switches the generator into the normal mode, which lets you edit the shape of the oscillator. This is especially advantageous for low-frequency oscillators, where the shape matters even though it doesn't have any physical meaning.

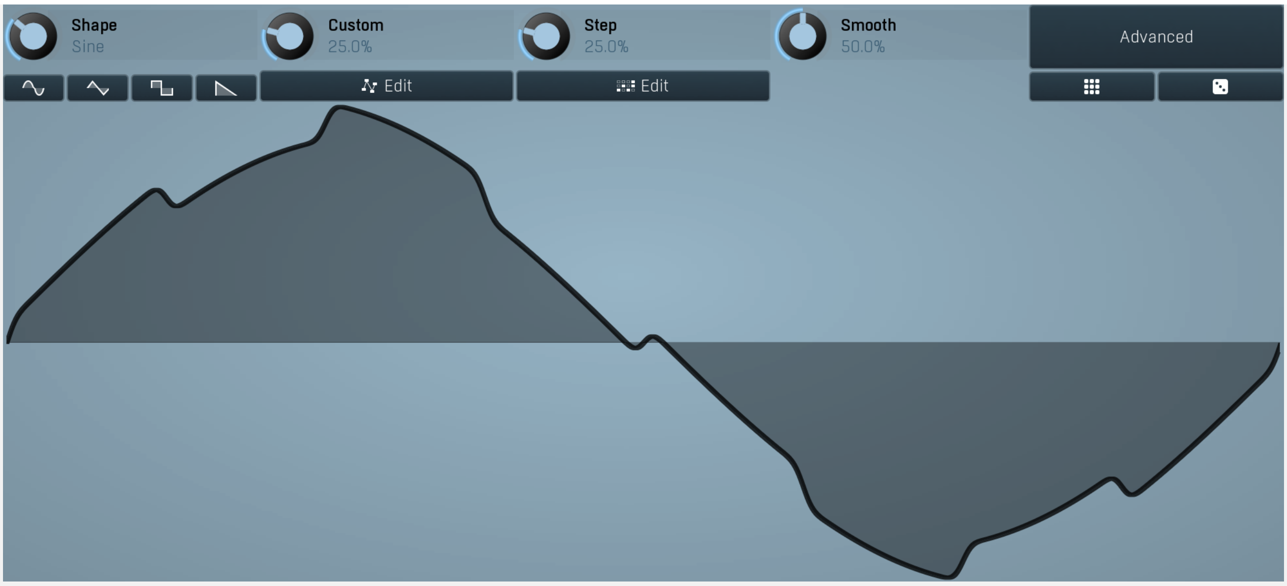
Convert

Convert button converts the current shape into harmonic-based representation. Please note that since the number of harmonics is limited, the result will not perfectly resemble the original shape.

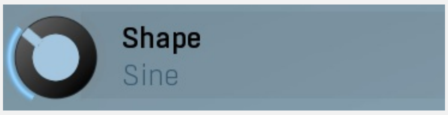
Harmonics

Harmonics button switches the generator into the harmonics mode, which lets you edit the levels and phases of individual harmonics. This is especially advantageous for high-frequency oscillators, hence sound generators.

Signal generator in Normal mode

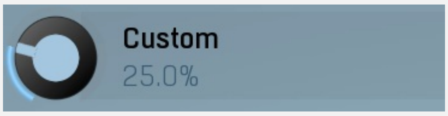


Signal generator in Normal mode works by generating the oscillator shape using a combination of several curves - a predefined set of standard curves, custom shape, step sequencer and custom sample. It also post-processes the shape using several filters including smoothing to custom transformations. This is especially useful when using the oscillator as an LFO (low-frequency-oscillator), where the harmonic contents does not really matter, but the shape does.



Shape

Shape controls the main shape used by the signal generator. There are several predefined shapes: exponential, triangle, sine power 8, sine power 4, sine square, sine, harmonics, more harmonics, disharmonics, sine square root, sine 4 root, rectangle, rect-saw, saw, noise and mess. You can choose any of them or interpolate between any 2 adjacent shapes using this control.

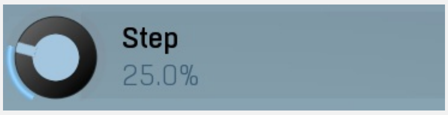


Custom

Custom controls the amount of the custom shape that is blended into the main shape.

Edit

Edit button shows the custom shape editor.

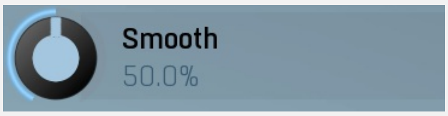


Step

Step controls the amount of the step sequencer shape that is blended into the main shape (which has already been blended with the custom shape).

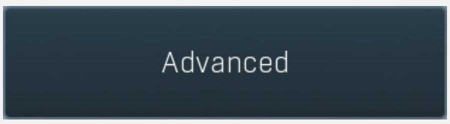
Edit

Edit button shows the step sequencer editor.



Smooth

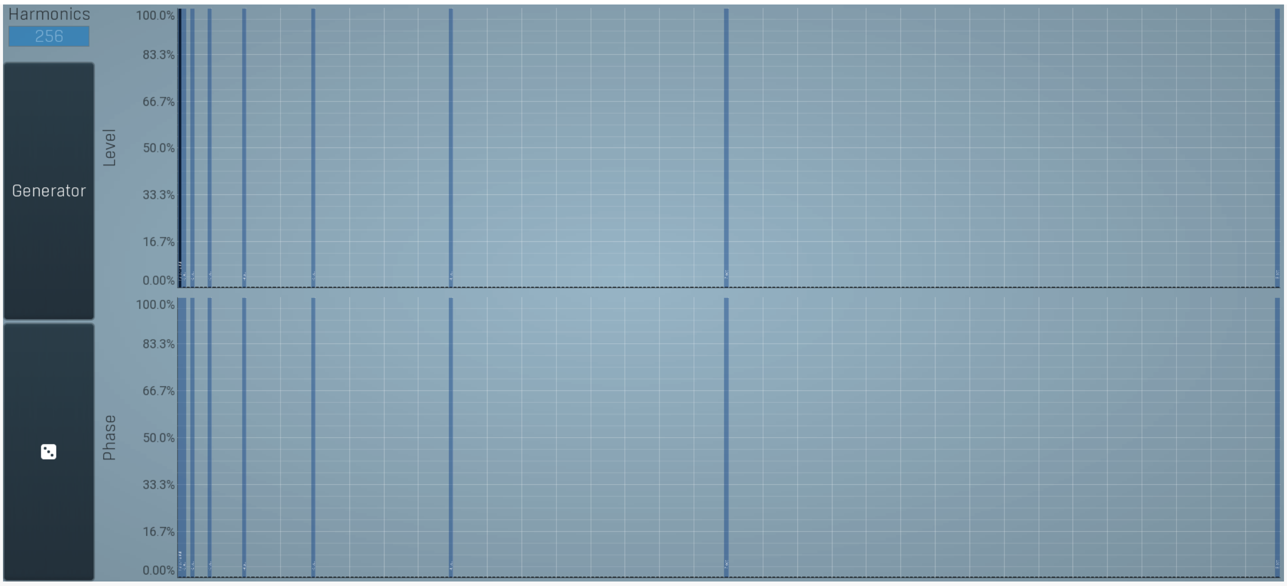
Smooth controls the amount of smoothing. Many shapes, especially those produced by the step sequencer, have rough jagged edges, which may be advantageous, but when used to modulate certain parameters, the output may be clicking or causing other artifacts. Smoothness helps it by smoothing the whole signal shape out and removing these rough edges.



Advanced

Advanced button displays an additional window with more advanced settings for post-processing the signal shape, such as harmonics or custom transformations.

Signal generator in Harmonics mode



Signal generator in Harmonics mode works by generating the oscillator shape using individual harmonics. Essentially a harmonic is a sine wave. The first harmonic, known as the fundamental, fits once in the oscillator time period, hence it is the same as selecting sine wave in the **Normal mode**. The second harmonic fits twice, the third three times etc. In theory, any shape you create in normal mode can be converted into harmonics. However, this approach to signal generation needs an enormous number of harmonics, which is both inefficient to calculate and mostly hard to edit. Therefore, the harmonic mode can process up to 256 harmonics, which is enough for very complex spectrums, however it is still not enough to generate an accurate square wave for example. If your goal is to create basic shapes, it is better to use the normal mode.

It is nearly impossible to say how a particular curve will sound when used as a high-frequency oscillator in a synthesizer, just by looking at its shape. Harmonics mode, on the other hand, is directly related to human hearing and makes this process very simple. In general, the more harmonics you add, the richer the sound will be. The higher the harmonic, the higher the tone. Usually, one leaves the first harmonic enabled too, as this is the fundamental tone, however you may experiment with more dissonant sounds without it.

Editing harmonics can be time consuming unless you hear what you want, so a signal generator is also available. This great tool lets you generate a random spectrum by a single click. You can also open the **Generator** settings and edit its parameters, which basically control the audio properties in a more natural way - using parameters such as complexity, harmonicity etc.

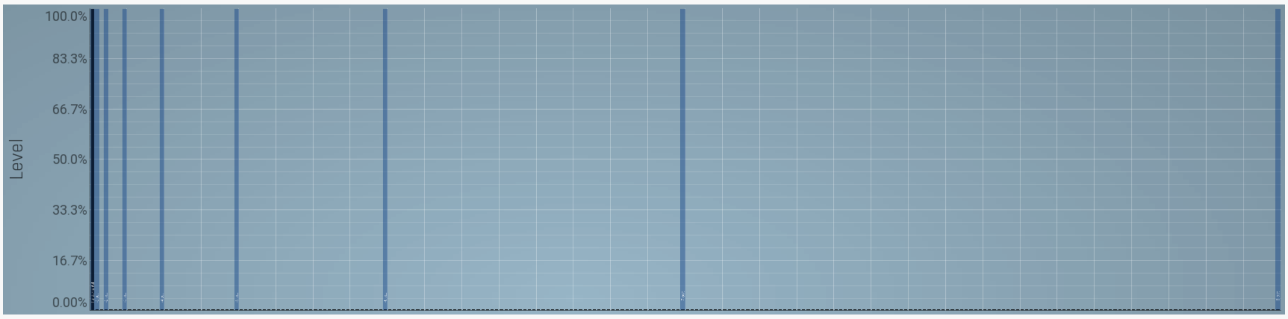
Generator

Generator button shows a powerful harmonics generator, which can create unlimited number of various timbres and even analyze a sample and extract harmonics from it.

Randomize

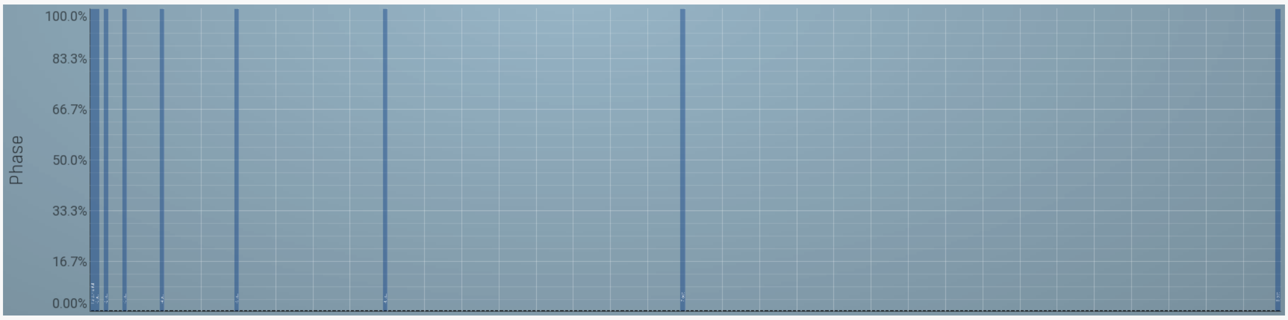
Randomize button selects random parameters for the harmonics generator, so you can use it to get a random sound character instantly.

Hold **Ctrl** to slightly modify existing generator settings instead of completely changing them.



Magnitudes graph

Magnitudes graph contains the levels of the individual harmonics. The highlighted bars are octaves, thus the 1st, 2nd, 4th, 8th harmonic etc.



Phases graph

Phases graph contains the phases of the individual harmonics. The highlighted bars are octaves, thus the 1st, 2nd, 4th, 8th harmonic etc.

Osc 2/3 panel

Osc 2/3 panel contains settings for the secondary oscillator. There are 2 secondary oscillators, neither of them has the unison feature, however these 2 have combination mode instead. By default the oscillator is simply mixed with the current signal, however the mode lets you perform somewhat more complicated transformation, such as frequency modulation or convolution.

Presets

Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.

Left arrow

Left arrow button loads the previous preset.

Right arrow

Right arrow button loads the next preset.

Randomize

Randomize button loads a random preset.

Copy

Copy button copies the settings onto the system clipboard.

Paste

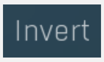
Paste button loads the settings from the system clipboard.

Random

Random button generates random settings using the existing presets.

Mode panel

Mode panel contains parameters controlling how the generator is used.



Invert

Invert switch inverts the oscillator output. This may be handy if multiple oscillators are cancelling each other.

Mode

Mode controls the way the generator is used.
Mix mode simply mixes the 2 signals, main input and modulator. Here, as below, the main input is the audio from the previous section and the modulator is the audio from this oscillator.

Frequency modulation performs a frequency modulation of main input signal by the modulator. This is essentially an extremely fast vibrato, which produces a very rich spectrum of harmonic and non-harmonic frequencies. Depth parameter controls the range of the vibrato, so the higher the depth is, the stronger the modulation. In effect, the depth parameter has the same effect as level of the modulator signal.

Frequency modulation (abs) is similar to the frequency modulation, but it provides an even richer spectrum and has a special feature - if the modulation signal is 0 (silence), the output is not changed at all. This is not true for the traditional frequency modulation, which takes effect even if the input is zero.
This may be useful when the modulation signal is temporary, mostly silent, and you don't want the input altered at all in the silent parts. A typical example is click simulation - most keyboard instruments "click" when a key is pressed. The click is very rich in spectrum and temporary obviously, which makes the frequency modulation (abs) an ideal candidate.

Ring modulation multiplies the 2 signals together, which results in so-called ring modulation effect, which usually provides a very rich output spectrum, however in most cases it is not harmonic any more, often sounding metallic. It is often a good idea to make one of the generators produce very low frequencies when using ring modulation. Depth parameter serves as a dry/wet ratio between the main input signal and output of the ring modulation.
The basic principle is that when you ring-modulate 2 sine waves with frequencies F1 and F2, the output will contain frequencies F1+F2 and F1-F2. It's therefore useful to have one of the frequencies (usually F2, called the modulator) low enough, even say 20Hz. That way the resulting frequencies will lie close to the original one and may enrich the signal.
An interesting thing happens when the 2 frequencies are an octave (or multiple octaves) apart. For example, if F1 is 440Hz (A4) and F2 is 220Hz (A3), then the resulting frequencies are 440-220=220Hz (again A3) and 440+220=660 (E5), which is nothing else than the 3rd harmonic. In this case the output of the ring modulation is still harmonic and provides an interesting way to enrich the

signal. In practice you'll like to use more complex waveforms than sine waves, which leads to quite more complicated results.

Frequency shifting is similar to ring modulation, but it performs some complicated filtering to remove the lower output frequency, F1-F2. In practice this is not completely possible, but it significantly attenuates the lower frequencies making the output cleaner. The name, frequency shifting is descended from the sole purpose - shifting all frequencies upwards or downwards. This is a quite different output from pitch shifting as it doesn't keep harmonic relationships between frequencies within the signal.

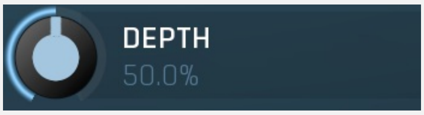
Maximization takes the maximum of both signals. This usually causes some form of dirtiness or distortion and in a way contains at least all frequencies from both signals. Hence this is yet another way to enrich them. Depth parameter serves as a dry/wet ratio between the main input signal and output of the maximization.

Convolution is inspired by traditional convolution and produces similar results. Whereas other modes make the signal richer by combining the spectral components of both signals in some way, convolution does the opposite - it enhances the spectral components present in both and attenuates those which are present in just one of the input signals. In other words, it filters one signal by another. Usually you will want to process a rich signal with another rich signal with this mode. Depth parameter controls the convolution length, which basically defines how deep the modulation will be. Note that the higher the depth, the more CPU power is needed.

Sync simulates hard synchronization - it resets the modulator every time the main input starts a new period. In this case the main input and modulator are swapped. That way the modulator controls the output spectrum, while main input controls the pitch. This kind of modulation provides reasonable results only for simple waveforms, the best start would be any simple wave (square, triangle, sine...) as main input and saw modulator. Then use modulator pitch and shape to change the sound character.

Dry/Wet

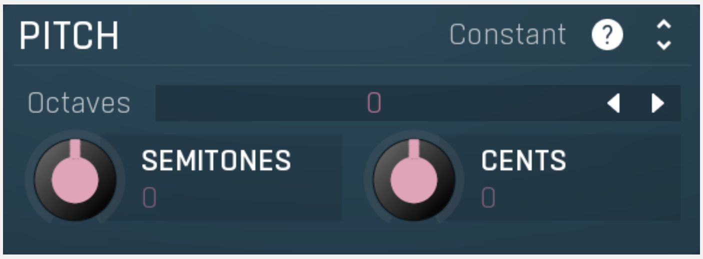
Dry/Wet defines the mix ratio for this oscillator.
Range: 0% wet, 100% dry to 100% wet, 0% dry, default 100% wet, 100% dry



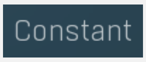
Depth

Depth controls the amount of modulation caused by this generator. It is used in some modes only.
Range: 0.00% to 100.0%, default 50.0%

Pitch panel



Pitch panel lets you shift the generator pitch.

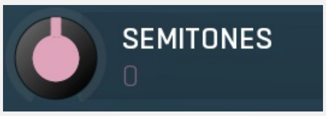


Constant frequency

Constant frequency makes the oscillator ignore the note pitch and behave as if an A4 (440Hz) note is received each time.

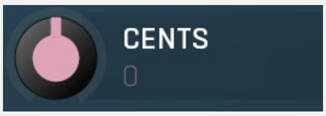
Octaves

Octaves defines the generator pitch change (in octaves).
Range: -8 to +8, default 0



Semitones

Semitones defines the generator pitch change (in semitones).
Range: -24.00 to +24.00, default 0



Cents

Cents defines the generator pitch change (in cents of a semitone). The actual pitch change is the sum of these 3 control values.
Range: -100.00 to +100.00, default 0

Transform panel

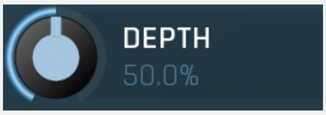
Transform panel contains some additional oscillator transformation features.

Mode

Mode controls the type of transformation used on the oscillator waveform. The oscillators themselves are pre-processed in a very complex manner to provide maximum audio quality. However that means that every time you change something in the oscillator editor, lots of calculations need to be performed, which may take quite some CPU. The transformations however are performed without pre-processing, so they may provide lower audio quality, but they are much less CPU intensive and often provide actually higher quality when modulated, because these are always sample accurate.

It is therefore recommended to use the oscillator editor for static sounds and then use transformations for modulation. Another reason to use the transformation is the actual harmonic properties of the generated sound - any oscillator (algorithmic or wavetable) produces only a fundamental frequency and its harmonics, the actual wave shape only controls the levels and phases of these frequencies. This means that if the fundamental pitch is say 100Hz, then the generated sound contains only 100Hz, 200Hz, 300Hz etc. It can never produce 150Hz for example. And that's when the transformations comes in handy again. When you transform the waveform in any way the result is just another waveform, so it still cannot produce other frequencies. However if you change the transformation depth in real-time, say using modulators, the output could produce just about any frequencies, all depending on the actual transformation and the way it is modulated.

Disabled disables any transformation.
PWM implements several pulse-width modulation algorithms, all affecting the width of the pulse or bending its shape in time. PWM tends to produce frequencies around each harmonic when modulated. The quicker the PWM depth is changed, the further these frequencies are from the main harmonics and the sound gets richer until a point where it is completely disharmonic.
Sync essentially duplicates the periods of the waveform (not necessarily a multiple). The higher the depth, the higher the dominant harmonic, so moving the depth in a way controls the pitch without losing the sense of the actual fundamental pitch. Sync waveform has abrupt edges in most cases. These generate lots of higher harmonic content, sometimes even reaching zipper noise. To diminish this and to get a cleaner sound, use the windowed versions, which smooth out the edges.
ASync transformations are similar to Sync, however the waveforms are bent, so the typical sync screaming is less resonant and more inharmonic.
Bending and mirroring alter the waveform quite smoothly, so these produce less rich tones and the effects may remind you of phasing. Discontinuities may occur when modulated quickly, so you can exploit that to get a richer sound if needed.
Reversing merges the waveform with a reversed replica of itself. This may often cause abrupt edges producing a sort of a modulated square wave. To get the most out of it, the original waveform should not be (anti)symmetric - when you look at a sine wave for example, the right half is just an inverted left half, so reversing it isn't such a miraculous action. This is true for most basic waves including saw and rectangle for example. Therefore reversing is most interesting when used with some more complex wave-shapes produced by harmonic mode or the step sequencer etc.
Invert, Maximize, Zero under, Power and Quantize are manipulating the amplitude only and due to the discontinuities these produce fairly rich sounds, lo-fi in a way. Please note the processor may produce different results for different quality modes, because it can generate specific waveforms for different notes to improve quality, but since the amplitude will be different, the results of the transformation will be different as well.
AM provides amplitude modulation with itself or a reversed version of itself. As such it provides minimum non-harmonic frequencies.
Recursive transformations are the most complex and basically transform the waveform by projecting it onto itself. The output generally depends on the input waveform signal and you might say that the more complex the initial waveform is, the more complex the output waveform will be. Recursive transformations may produce extremely full spectra and may mix the properties and richness of other transformations combined. As such, simple waveforms such as sine and similar usually produce the best results. Modulating a harmonically complex waveform can easily end up with white noise.



Depth

Depth controls the amount/type of selected transformation and shall be used for modulation. Its meaning depends on the selected transformation.
Range: 0.00% to 100.0%, default 50.0%

Panorama

Panorama defines the oscillator panorama transformation.
Range: 100% left to 100% right, default center

Shape graph editor

Shape graph editor controls the oscillator shape. You can use either oscillator shapes or harmonics; the latter are preferred in this case as they directly control the harmonic structure of the generated sounds. Note that displaying the output shape using an oscilloscope may produce very different images than what you see here. That is not a bug. It's the plugin ensuring it sounds as good as possible, analogue-like.Signal-generator is an incredibly versatile generator of low & high frequency signals. It offers 2 distinct modes - Normal and Harmonics.
Normal mode is appropriate for low-frequency oscillators, where the graphical shape is relevant and is used to drive some form of modulation. For example, a tremolo uses this modulation to change the actual signal level in time. Frequencies for such oscillators usually do not exceed 20Hz as this is a sort of limit above which the frequencies become audible.
Harmonics mode is designed for high-frequency oscillators, where the actual shape is not as important as the harmonic content of the resulting signal, hence it is especially useful for actual audio signals. Please note that since a shape can contain more harmonics than those available from the harmonic generator, the results may not be exactly the same. As an example, a rectangular wave in normal mode may sound fuller than when converted to the harmonic mode.

Use the arrow-down button to switch from normal mode to harmonics mode or click the **Normal** and **Harmonics** buttons

Normal mode

The generator first uses a set of predefined signal shapes (sine, triangle, rectangle...), which you can select directly by right-clicking on the editor and choosing the requested shape from the menu. This menu also provides a link to the modulator shapes preset manager, normalization and randomization. You can also use the **Main shape** parameter, which generates a combination of adjacent signals to provide a nearly inexhaustible number of basic shapes.

The engine then combines the predefined shape with a **Custom shape**, which may be anything you can draw using the advanced envelope engine, depending on the level set by the **Custom shape** control. Use the **Edit** button to edit the custom shape.

You can also combine those results with a fully featured step sequencer, with variable number of steps and several shapes for each of them, depending on the level set by the **Step sequencer** control. Use the lower **Edit** button to edit the step sequence.

Those results may be mixed with a custom sample, which is available from the advanced settings, accessed by clicking the **Advanced** button.

Smoothness softens any abrupt edges, generated by the step sequencer for example.

Finally there are **Advanced** features providing more complex transformations, adding harmonics etc. or you can click the **Randomize** button in the top-left corner to generate a random, but reasonable, modulator shape.

Harmonics mode

Harmonics mode represents the signal as a series of harmonics (that is, multiples of the base frequency). For example, when your oscillator has a frequency of 2Hz (set in the **Rate** panel), then the harmonics are 2Hz, 4Hz, 6Hz, 8Hz etc. In theory, any signal can be created by mixing a potentially infinite number of these harmonics.

The harmonics mode lets you control the levels and phases of each harmonic. The top graph controls the levels of individual harmonics, while the bottom one controls their phases. Use the left-mouse button to change the values in each graph, the right-mouse button sets the default for the harmonics - 0% level and 0% phase. In both graphs the harmonics of power 2 (that is octaves) are highlighted. Other harmonics may actually sound disharmonic, despite their names.

For example, if you reset all harmonics to the defaults and increase only the first one, you will get a simple sine wave. By adding further harmonics you make the output signal more complex.

Harmonics controls the number of generated harmonics. The higher the number is, the richer the output signal is (unless the levels are 0% of course). This is useful to make the sound cleaner. For example, if you transform a saw-tooth wave to harmonics, it would not sound like a typical saw-tooth wave anymore, but more like a low-passed version of one. The more harmonics you use, the closer you get to the original saw-tooth wave.

Generator is a powerful tool for generating the harmonics, which are otherwise rather clumsy to edit. The generator provides several parameters based upon which it creates the entire series of harmonic levels and phases. These parameters are usually easier to understand than the harmonics themselves. Part of the generator is the randomizer available via the **Random seed** button, which smartly generates random settings for the generator. This makes the process of getting new sounds as simple as possible.

Signal generation fundamentals

The signal generator produces a periodic signal with specified wave shape. This means that the signal is repeating over and over again. As a result it can only contain multiples of the fundamental frequency. For example, if the generator is producing 100Hz signal, then it can contain 100Hz (fundamental or 1st harmonic), 200Hz (2nd harmonic), 300Hz (3rd harmonic), 400Hz (4th harmonic) etc. However, it can never produce 110Hz. You can then control the level of each harmonic and their relative phases. It does not matter whether you use the normal mode using oscillator shapes, or harmonics mode where you can control the harmonics directly. If both modes result in the same wave shape (such as sine wave vs. 1st harmonic only), then the result is exactly the same.

Sine wave is the simplest of all as it contains the fundamental frequency only. The "sharper" the signal shape is, the more harmonics it contains. The biggest source of higher harmonics is a "discontinuity", which you can see in both rectangle and saw waves. In theory, these signals have an infinite number of harmonics. However since our hearing is highly limited to less than 20kHz, the number of harmonics which are relevant is actually pretty small. If you generate a 50Hz signal, which is very low, and assuming that you have extremely good ears and you actually hear 20kHz, then the number of harmonics audible for you is 20000 / 50 = 400.

What happens above 20kHz?

Consider the example above again, what happens with harmonics above 400? These either stay there and simply are not audible, disappear if anti-aliasing is used, or get aliased back under 20kHz in which case you get the typical digital dirt.

When you convert a rectangle wave to harmonics mode, only the first 256 harmonics are used, so it basically works like an infinitely steep low-pass filter. What is the limit then? 50 Hz * 256 = 12.8kHz. The harmonic mode will not produce anything above this limit if you are generating a 50Hz signal. Most people do not hear anything above 15kHz, so this is usually enough, but if not, you may need to use the normal mode where you get the "infinite" number of harmonics.

What you see is not always what you get!

Say you want a rectangle wave and play a 440Hz tone(A4). You would expect the output signal to be a really quick rectangle wave, right? Wrong! If you would do that, and actually most synthesizers on the market do that, you would get the infinite number of harmonics. And, since you are working in say 48kHz sampling rate, the maximum frequency that can actually exist in your signal is 24kHz. So everything above it would get aliased below 24kHz, and there would be a lot of aliased dirt.

The "good" synthesizers perform a so-called anti-aliasing. There are several methods, most of them require quite a lot of CPU or have other limitations. The goal is to remove all frequencies above the 24kHz in our case or in reality, it is more about removing all aliased frequencies above 20kHz - this means, that we do not care about frequencies above 20kHz, because we do not hear them anyway. But we will keep it simple. Let's say we remove everything above 20kHz. You already know that the rectangle wave can be created using an infinite number of harmonics or sine waves. We removed everything above the 45th harmonic (20000 / 440) so our rectangle wave is trying to be formed using just 45 harmonics, so it will not really look like a rectangle wave.

After some additional filtering (like DC removal), the rectangle wave may look completely different than a true rectangle wave, yet it would sound the same! Does it matter? Not really. You simply edit the shape as a rectangle wave and let the synthesizer do the ugly stuff for you. But do not check the output, because it may be very different than what you would expect ;).

How can I generate non-harmonic frequencies?

Ok, so now you are playing a 440Hz (A4) saw wave, it contains 440Hz, 880Hz, 1320Hz etc. Anything generated using the signal generator can contain only these frequencies, the only difference is the levels and phases of each of them. What if you want to make the signal dirty by adding say 500Hz? Well, that is not that simple! Here we are getting into audio synthesizer stuff, so let us just give you a few hints.

The traditional way is to use modulation. One particular method is called frequency modulation (FM). Instead of generating a 440Hz saw wave with your generator, you change the pitch, up and down. You are modulating the frequency, that's why FM. It is basically a vibrato, but as you increase the speed of the vibrato, it gets so quick that you stop noticing the pitch changes (that's very simplified but it serves the purpose) and instead it starts producing a very complex spectrum. Will the 500Hz be there? Well, if setup correctly, yes, but there will also be lots of other non-harmonic frequencies.

Another way is possible without any other tools. Let's say you do not want 440Hz, but 660Hz. Then you may generate 220Hz instead of 440Hz (which is one octave below it) and voila, 660Hz is the 3rd harmonic (3 x 220 is 660)! But you need to shift the saw wave one octave above. Fortunately it is not that hard here - go to the normal mode, select saw tooth, click advanced, and use the harmonics panel to remove the fundamental and leave just the 2nd harmonic, then convert it to harmonic mode. Well, it's not that hard, but it's not exactly simple either...

The only way is, of course, additive synthesis. In that case you do not use one oscillator, but many of them. It lets you generate just about anything. But there is a catch, actually many of them. First, you need to say "ok I want this frequency and that frequency...", the setup is actually infinitely hard as there may be an infinite number of frequencies :). And the second is, of course, CPU requirements.

So is there some ultimate solution? Nope, sorry. The good thing is, you will not probably need it, because while what you see is not always what you get, also what you want is often not what you really want to hear :).

Presets

Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.

Left arrow

Left arrow button loads the previous preset.

Right arrow

Right arrow button loads the next preset.

Randomize

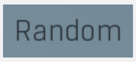
Randomize button loads a random preset.

Copy

Copy button copies the settings onto the system clipboard.

Paste

Paste button loads the settings from the system clipboard.



Random

Random button generates random settings using the existing presets.

Normal

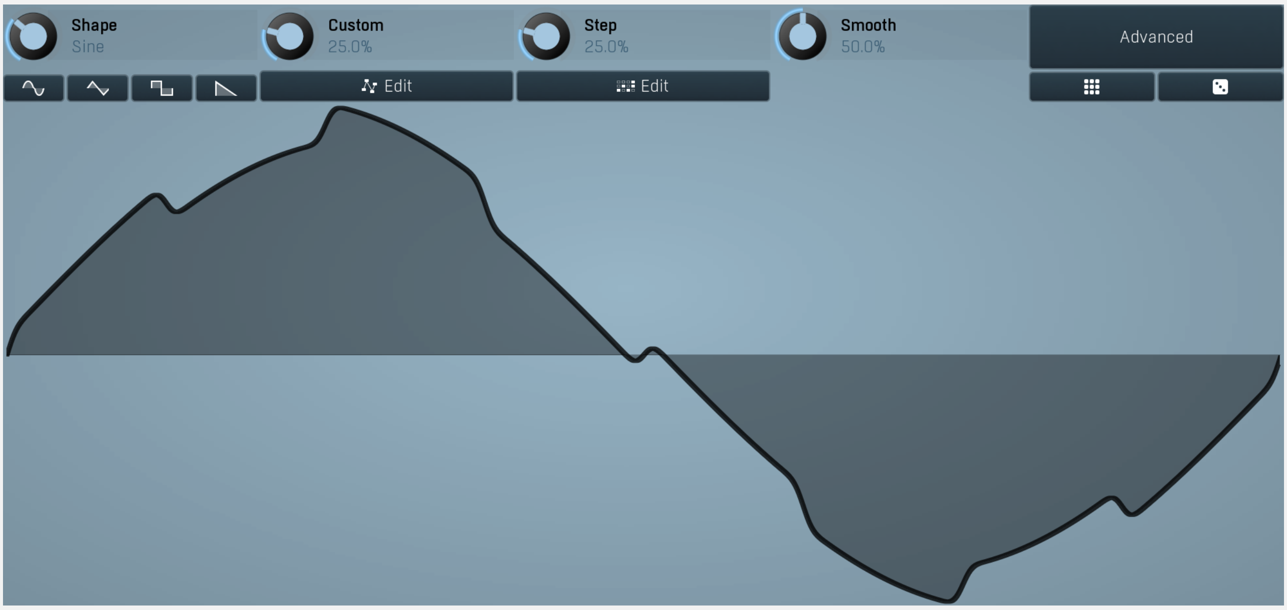
Normal button switches the generator into the normal mode, which lets you edit the shape of the oscillator. This is especially advantageous for low-frequency oscillators, where the shape matters even though it doesn't have any physical meaning.

Convert

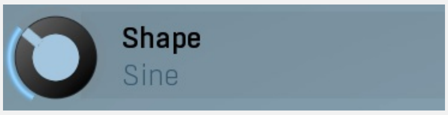
Convert button converts the current shape into harmonic-based representation. Please note that since the number of harmonics is limited, the result will not perfectly resemble the original shape.

Harmonics
Harmonics button switches the generator into the harmonics mode, which lets you edit the levels and phases of individual harmonics. This is especially advantageous for high-frequency oscillators, hence sound generators.

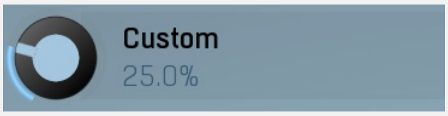
Signal generator in Normal mode



Signal generator in Normal mode works by generating the oscillator shape using a combination of several curves - a predefined set of standard curves, custom shape, step sequencer and custom sample. It also post-processes the shape using several filters including smoothing to custom transformations. This is especially useful when using the oscillator as an LFO (low-frequency-oscillator), where the harmonic contents does not really matter, but the shape does.

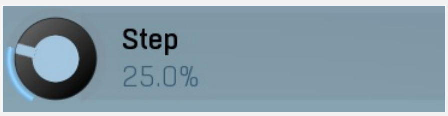


Shape
Shape controls the main shape used by the signal generator. There are several predefined shapes: exponential, triangle, sine power 8, sine power 4, sine square, sine, harmonics, more harmonics, disharmonics, sine square root, sine 4 root, rectangle, rect-saw, saw, noise and mess. You can choose any of them or interpolate between any 2 adjacent shapes using this control.



Custom
Custom controls the amount of the custom shape that is blended into the main shape.

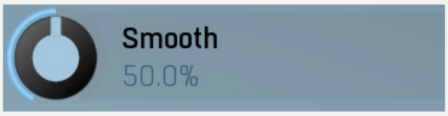
Edit
Edit button shows the custom shape editor.



Step
Step controls the amount of the step sequencer shape that is blended into the main shape (which has already been blended with the custom shape).

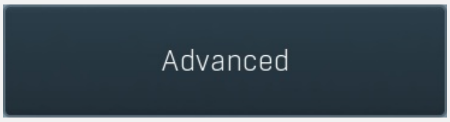
Edit

Edit button shows the step sequencer editor.



Smooth

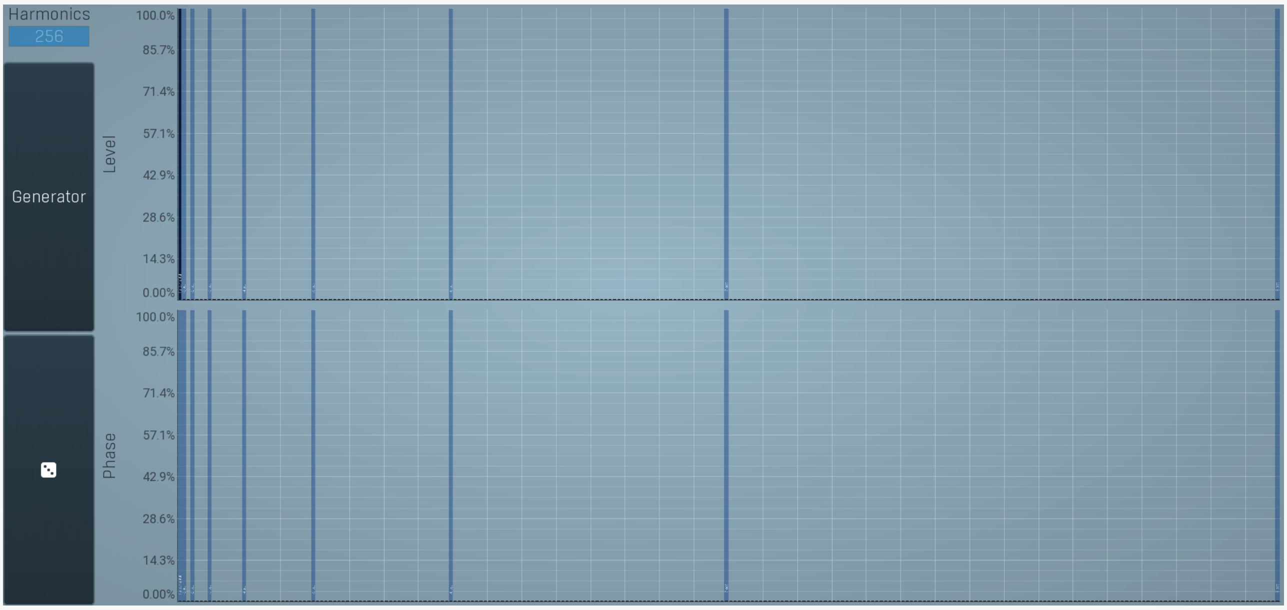
Smooth controls the amount of smoothing. Many shapes, especially those produced by the step sequencer, have rough jagged edges, which may be advantageous, but when used to modulate certain parameters, the output may be clicking or causing other artifacts. Smoothness helps it by smoothing the whole signal shape out and removing these rough edges.



Advanced

Advanced button displays an additional window with more advanced settings for post-processing the signal shape, such as harmonics or custom transformations.

Signal generator in Harmonics mode



Signal generator in Harmonics mode works by generating the oscillator shape using individual harmonics. Essentially a harmonic is a sine wave. The first harmonic, known as the fundamental, fits once in the oscillator time period, hence it is the same as selecting sine wave in the **Normal mode**. The second harmonic fits twice, the third three times etc. In theory, any shape you create in normal mode can be converted into harmonics. However, this approach to signal generation needs an enormous number of harmonics, which is both inefficient to calculate and mostly hard to edit. Therefore, the harmonic mode can process up to 256 harmonics, which is enough for very complex spectrums, however it is still not enough to generate an accurate square wave for example. If your goal is to create basic shapes, it is better to use the normal mode.

It is nearly impossible to say how a particular curve will sound when used as a high-frequency oscillator in a synthesizer, just by looking at its shape. Harmonics mode, on the other hand, is directly related to human hearing and makes this process very simple. In general, the more harmonics you add, the richer the sound will be. The higher the harmonic, the higher the tone. Usually, one leaves the first harmonic enabled too, as this is the fundamental tone, however you may experiment with more dissonant sounds without it.

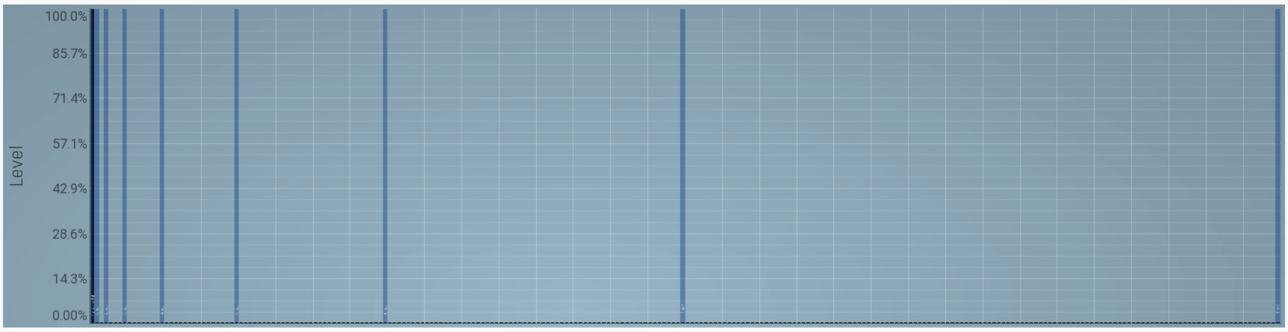
Editing harmonics can be time consuming unless you hear what you want, so a signal generator is also available. This great tool lets you generate a random spectrum by a single click. You can also open the **Generator** settings and edit its parameters, which basically control the audio properties in a more natural way - using parameters such as complexity, harmonicity etc.

Generator

Generator button shows a powerful harmonics generator, which can create unlimited number of various timbres and even analyze a sample and extract harmonics from it.

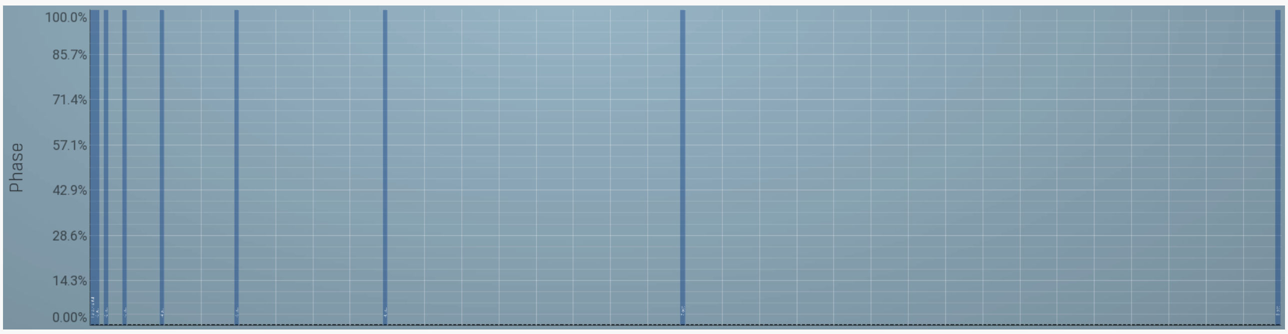
Randomize

Randomize button selects random parameters for the harmonics generator, so you can use it to get a random sound character instantly. Hold **Ctrl** to slightly modify existing generator settings instead of completely changing them.



Magnitudes graph

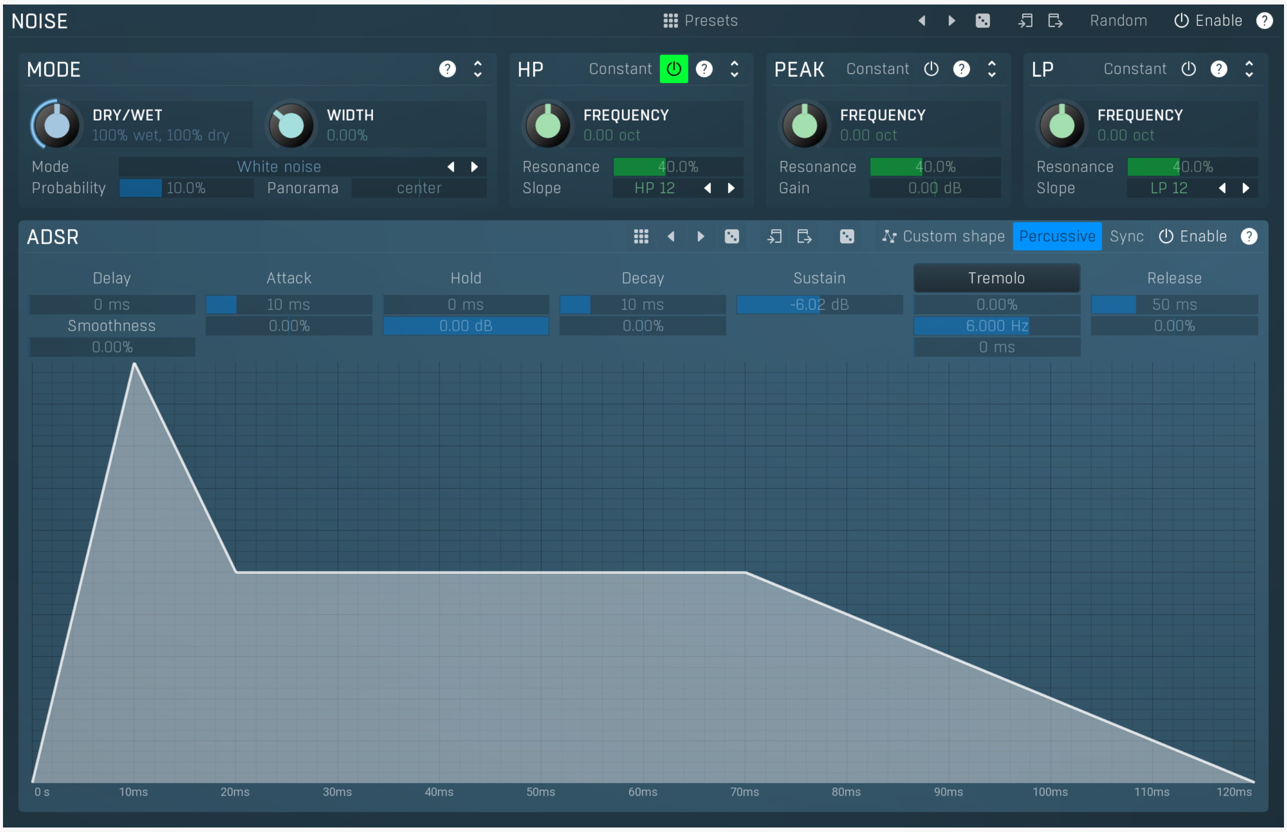
Magnitudes graph contains the levels of the individual harmonics. The highlighted bars are octaves, thus the 1st, 2nd, 4th, 8th harmonic etc.



Phases graph

Phases graph contains the phases of the individual harmonics. The highlighted bars are octaves, thus the 1st, 2nd, 4th, 8th harmonic etc.

Noise panel



Noise panel contains parameters of the noise generator, which can be mixed with the main signal. The noise generator can also have its own ADSR envelope.

Presets

Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.

Left arrow

Left arrow button loads the previous preset.

Right arrow

Right arrow button loads the next preset.

Randomize

Randomize button loads a random preset.

Copy

Copy button copies the settings onto the system clipboard.

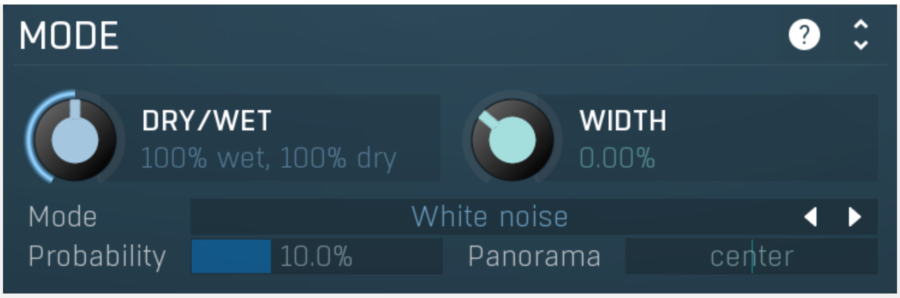
Paste

Paste button loads the settings from the system clipboard.

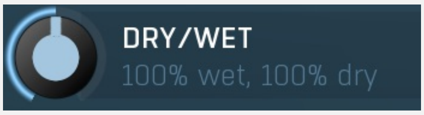
Random

Random button generates random settings using the existing presets.

Mode

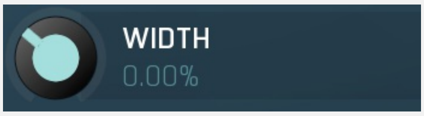


Mode contains the overall noise generator parameters.



Dry/Wet

Dry/Wet defines the mix ratio for the noise generator.
Range: 0% wet, 100% dry to 100% wet, 0% dry, default 100% wet, 100% dry



Width

Width controls the noise stereo width.
Range: Mono to 200.0%, default 0.00%

Mode

Mode defines the type of output randomized signal. **White noise** generates standard white noise, which has equal energy in each frequency. **Pink noise** provides an accurate approximation of pink noise, where the energy is decreasing by 3dB per octave. **Clicks & pops** define generated random peaks with specified probability. **Stairs** is similar to clicks & pops, but after each peak the generated value is following instead of generating zeros. Hence, the resulting waveform looks like random stairs.

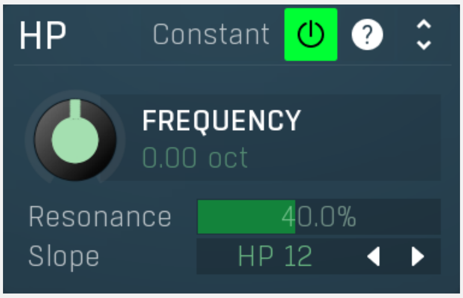
Probability

Probability defines a probability coefficient used in some of the modes somehow. For example for clicks & pops mode it affects how often the clicks happen.
Range: 0.00% to 100.0%, default 10.0%

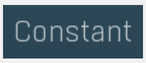
Panorama

Panorama defines the noise generator panorama transformation.
Range: 100% left to 100% right, default center

HP panel



HP panel contains parameters of the noise generator high-pass filter.



Constant frequency

Constant frequency makes the filter ignore the note pitch and behave as if an A4 (440Hz) note is received each time.

Frequency

Frequency controls the high-pass filter frequency.
Range: -8.00 oct to +8.00 oct, default 0.00 oct

Resonance

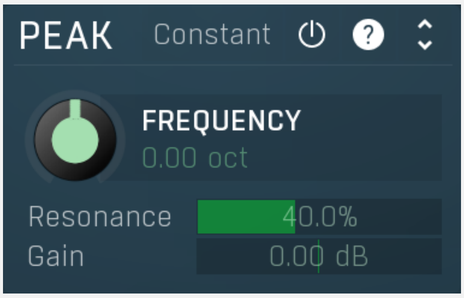
Resonance controls the high-pass filter resonance.

Range: 0.00% to 100.0%, default 40.0%

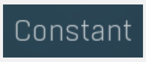
Slope

Slope controls the high-pass filter slope.

Peak



Peak contains parameters of the noise generator peak filter.



Constant frequency

Constant frequency makes the filter ignore the note pitch and behave as if an A4 (440Hz) note is received each time.

Frequency

Frequency controls the peak filter frequency.
Range: -8.00 oct to +8.00 oct, default 0.00 oct

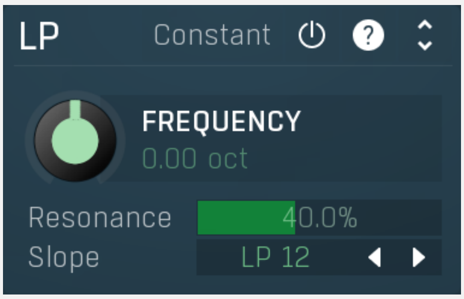
Resonance

Resonance controls the peak filter resonance.
Range: 0.00% to 100.0%, default 40.0%

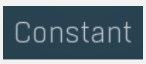
Gain

Gain controls the peak filter gain.
Range: -48.00 dB to +48.00 dB, default 0.00 dB

LP



LP contains parameters of the noise generator low-pass filter.



Constant frequency

Constant frequency makes the filter ignore note pitch and behave like A4 (440Hz) has always been received.

Frequency

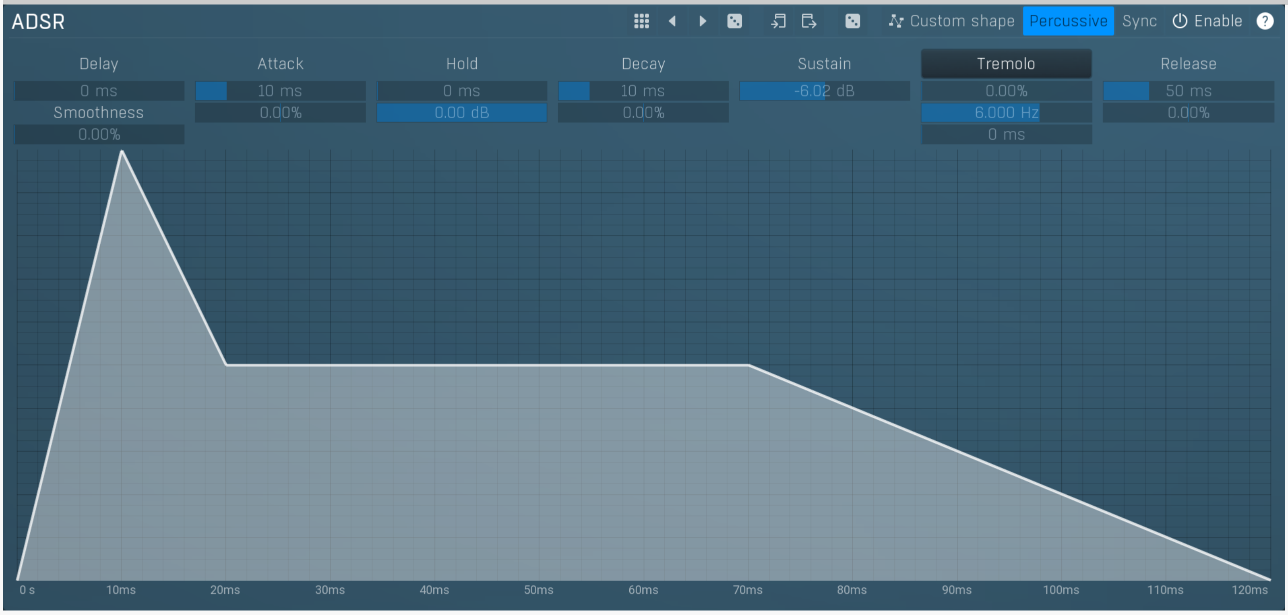
Frequency controls the low-pass filter frequency.
Range: -8.00 oct to +8.00 oct, default 0.00 oct

Resonance

Resonance controls the low-pass filter resonance.
Range: 0.00% to 100.0%, default 40.0%

Slope

Slope controls the low-pass filter slope.



ADSR graph

ADSR graph controls the ADSR envelope for the noise generator.Our ADSR envelopes are much more sophisticated than classic attack-decay-sustain-release envelopes. Besides these common parameters they also let you control the curvature of each stage. Additionally, there are hold and delay sections ("DAHDSR"), global smoothing and tremolo. You can even use the custom shape mode to define your own attack/release curves.

Presets

Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.

Left arrow

Left arrow button loads the previous preset.

Right arrow

Right arrow button loads the next preset.

Randomize

Randomize button loads a random preset.

Copy

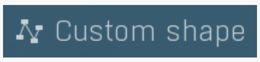
Copy button copies the settings onto the system clipboard.

Paste

Paste button loads the settings from the system clipboard.

Random

Random button generates random settings using the existing presets.



Custom shape

Custom shape button enables custom shape mode, which lets you draw your own attack and release stages using the envelope system. Both stages are then automatically connected to form the resulting envelope.

Percussive

Percussive button activates the immediate release mode in which case the note-off causes an immediate switch to the release stage. If this is disabled, the release stage does not occur until the whole attack/decay stage finishes.

Sync

Sync button controls the ADSR tempo sync feature. By default this is disabled and means that all times are followed exactly, meaning that if **Attack** is say 100ms, then it will be 100ms indeed. Tempo sync lets the plugin adjust the times to ensure it will be always in sync with the host tempo. In this case 100ms may become say 125ms if the tempo is 120bpm, because 125ms is the length of a 16th note. This makes it extremely simple to convert any envelope to a tempo-synced one. The plugin always chooses the nearest longer note, in other words it always round up.
Straight and **Triplets** modes automatically find 'nice' values.
For example, if a 16th note takes 100ms, the attack time is 550ms, and the sync mode is straight, then the plugin checks for 100ms, find out that it is too low, so it checks 8th note, being 200ms, still too low, then continues with quarter note, which takes 400ms, and still not enough, finally 800ms corresponding to a half note is the one, so the resulting time will be 800ms. Triplet cases are more complex, but the principle is the same.
1/16, **1/8** and **1/4** modes choose the nearest higher multiply of the base note length. For example, if a 16th note takes 100ms, the attack time is 550ms, and the sync mode is 1/16, the resulting time will be 600ms.

Tremolo

Tremolo button displays additional tremolo settings, containing tremolo behaviour and shape.

Delay

Delay lets you shift the entire envelope forwards in time. While this doesn't make much sense for a global instrument envelope for instance, it may be well useful to control characteristics of evolving sounds.

Attack

Attack controls the length of the initial stage of the envelope. It is one of the most important parameters controlling how quick the initial transient is. For most instruments the length is quick short, but for pads and other slowly evolving sounds it is quite common to set this to several seconds.

Hold

Hold specifies the time the level stays at maximum after the attack stage.

Decay

Decay controls the time it takes for the level to drop from the maximum to the **Sustain**. If the sustain is 0dB, then this parameter has no effect, because in a way the sustain stage starts immediately after the attack.

Sustain

Sustain controls the sustain level. For most sounds the initial attack transient is the highest point of the entire sound. Imagine playing a string instrument, such as a guitar, the initial hit to the strings is represented by the attack+hold+decay sections and is the most prominent. After that the level drops to the sustain stage, where it holds for most of the time.

Tremolo

Tremolo defines the amount of the tremolo effect that is engaged in the sustain, or even in the decay section and continues until the envelope ends. While this is a rather unusual feature for an envelope to have, it is very handy for simulating various effects human players do when performing on real instruments, such as the tremolo or vibrato.

Release

Release controls the length of the release section, which usually starts when a note is released.

Attack shape

Attack shape controls the shape of the attack section and defines its sound character.

Hold level

Hold level controls the level of the hold section. By default it equals maximum meaning that the hold section actually holds the maximum level. However by making it lower you can sort of simulate 2 separate decay sections, first going from maximum to hold level, second going from hold level to sustain.

Decay shape

Decay shape controls the shape of the decay section and defines its sound character.

Tremolo rate

Tremolo rate controls the speed of the tremolo. In the tremolo settings it is possible to control additional characteristics including tempo sync.

Release shape

Release shape controls the shape of the release section and defines its sound character.

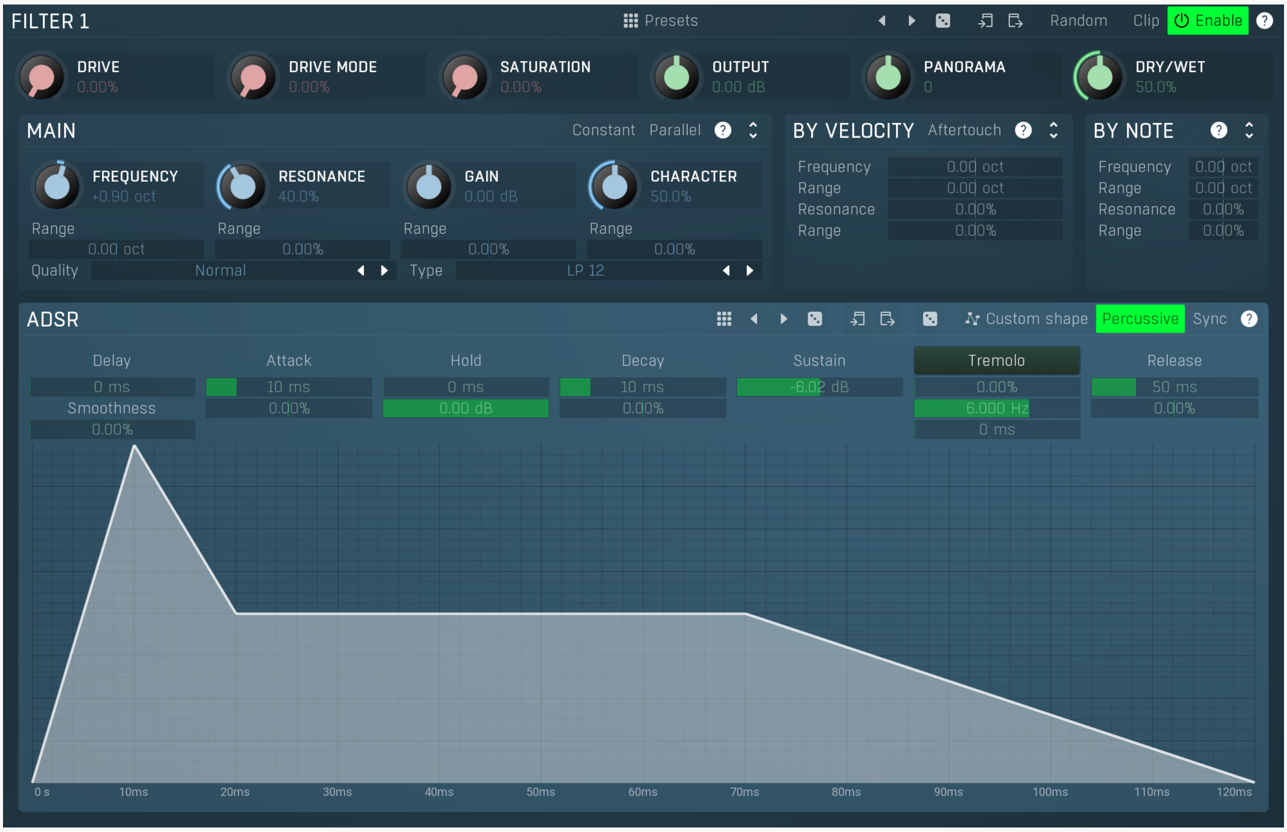
Smoothing

Smoothing lets you smoothen the entire envelope avoiding abrupt jumps. Note that in some cases involving short jumps the results may be a bit obscure.

Tremolo fade-in

Tremolo fade-in defines the time for the tremolo to reach its full level. It is a natural behaviour of human players (on say a saxophone) that they don't start a full tremolo immediately and rather let the modulation rise to maximum over a period of time.

Filter panel



Filter panel contains settings for the particular filter.

Presets

Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.

Left arrow

Left arrow button loads the previous preset.

Right arrow

Right arrow button loads the next preset.

Randomize

Randomize button loads a random preset.

Copy

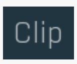
Copy button copies the settings onto the system clipboard.

Paste

Paste button loads the settings from the system clipboard.

Random

Random button generates random settings using the existing presets.



Clip

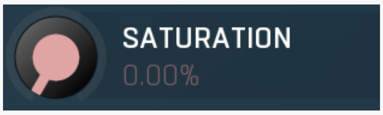
Clip enables an optional clipper placed after the filter's output gain. It can be used to remove potential peaks that the filter may cause especially with wild settings. As a hard clipper it can also be used as another distortion stage, however be cautious with it.

Drive

Drive controls the input distortion of the filter. This creates higher harmonics, which are then processed together with the original signal through the filter. Usually the drier the input signal is, the more drive may be used to make the signal richer before any filtering occurs. When applied to an already-rich signal, the results may simply be too dirty. However when applied to a rich yet harmonic signal (such as a saw-tooth wave), only existing harmonics will be added, so the effect won't be creating additional harmonics but rather changing their levels resulting in a different spectrum. It is highly advised to use the plugin's oversampling feature in order to minimize disharmonic components created by aliasing.
Range: 0.00% to 100.0%, default 0.00%

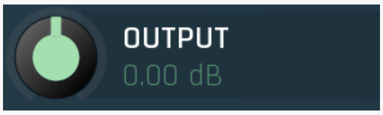
Drive mode

Drive mode controls the input distortion character. Essentially this controls the levels of different harmonics.
Range: 0.00% to 100.0%, default 0.00%



Saturation

Saturation controls the output saturation that is performed before the output gain. This provides further enrichment performed after the filtering. *For example, you may have a simple sine wave on the input, processed through the input distortion, which adds several harmonics. A filtering using a low-pass filter may then remove most of the higher harmonics content. Saturation may then be used to generate some of the harmonics back again.*
Range: 0.00% to 100.0%, default 0.00%



Output gain

Output gain defines the output gain that is applied after the filter. This could be useful for controlling the input to the next stages - the next filter, global saturation. The rule of thumb is - the higher the **Drive** or filter **Gain**, the lower this output gain should be to compensate. *For example, you may set a high drive followed by a Sub-X with high gain in the first filter. Its input distortion will generate lots of higher harmonics as well as change the output level, and the filter would increase the level even more. This could easily be more than +20dB, which when fed to the following filter's distortion or global saturation, may be unusable as each of these nonlinear processors would be immediately overdriven. Just use the filter's output gain to compensate for this by dropping it down.*
Range: -48.00 dB to +48.00 dB, default 0.00 dB

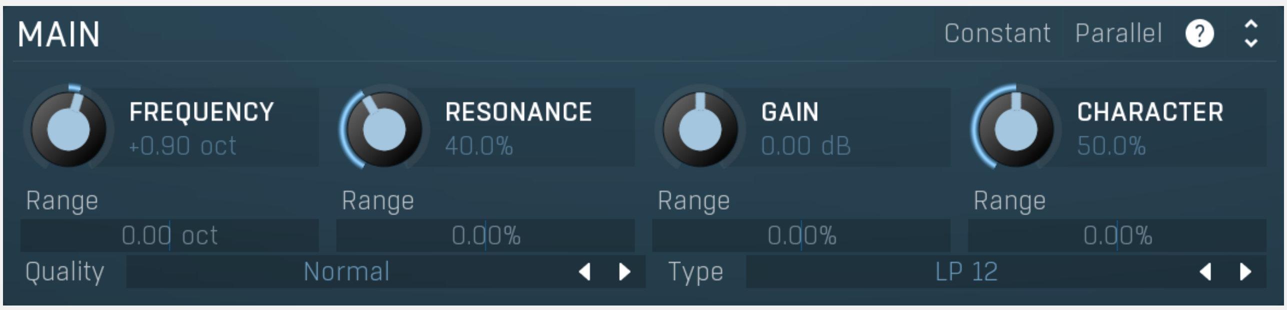
Panorama

Panorama lets you shift the filter frequency between channels. Left channel's frequency is shifted down by specified amount, right channel is shifted up, third down etc.
Range: -48.00 to +48.00, default 0

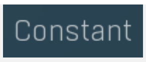
Dry/Wet

Dry/Wet controls the filter dry/wet ratio. Please note that since most filters are altering phase, you may experience various frequency level changes for values other than 0% and 100%.
Range: 0.00% to 100.0%, default 100.0%

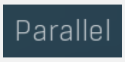
Main panel



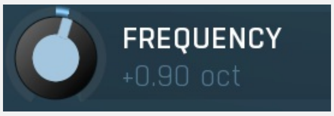
Main panel contains the main filter parameters.



Constant frequency
Constant frequency makes the filter ignore the note pitch and behave as if an A4 (440Hz) note is received each time. You can still use the velocity/note tracking when this is enabled.



Parallel
Parallel enables parallel processing for the filters. Normally each filter processes the output of the previous filter and its output becomes the new signal. If parallel processing is enabled, all filters process the original generator signal instead and their outputs are mixed together.

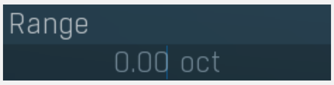


Frequency
Frequency controls the filter minimum frequency.
Range: -8.00 oct to +8.00 oct, default 0.00 oct

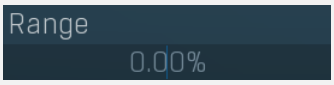
Resonance
Resonance controls the filter central resonance. Please note that it is used only for some filter types.
Range: 0.00% to 100.0%, default 40.0%

Gain
Gain controls the central gain of the filter. Please note that it is used only for some filter types.
Range: -48.00 dB to +48.00 dB, default 0.00 dB

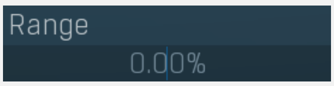
Character
Character controls the central character of the filter. Please note that it is used only for some filter types. Character affects some additional filter specific features, such as dispersion of harmonics. For polymorph filters character actually controls the internal structure of the filter and any change to this value completely changes the algorithm providing maximum unique sound combinations. Therefore character modulation is not available for polymorph filters, as modulation would create extreme changes.
Range: 0.00% to 100.0%, default 50.0%



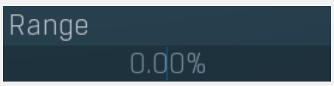
Frequency range
Frequency range controls the extent to which the frequency is modulated. With 0% no modulation occurs and the filter frequency is defined by the **Frequency** parameter only. If you increase the range however, the filter frequency will move within the range frequency ... frequency + range according to the ADSR envelope.
Range: -8.00 oct to +8.00 oct, default 0.00 oct



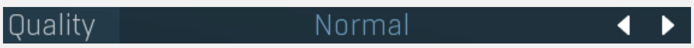
Resonance range
Resonance range controls the extent to which the resonance is modulated. With 0% no modulation occurs and the filter resonance is defined by **Resonance** parameter only. If you increase the range however, the filter resonance will move away from the central resonance value according to the ADSR envelope.
Range: -100.0% to 100.0%, default 0.00%



Gain range
Gain range controls the extent to which the gain is modulated. With 0% no modulation occurs and the filter gain is defined by the **Gain** parameter only. If you increase the range however, the filter gain will move away from the central gain value according to the ADSR envelope.
Range: -100.0% to 100.0%, default 0.00%



Character range
Character range controls the extent to which the character is modulated. With 0% no modulation occurs and the filter character is defined by **Character** parameter only. If you increase the range however, the filter character will move away from the central character value according to the ADSR envelope.
Range: -100.0% to 100.0%, default 0.00%



Quality
Quality controls the ratio between audio quality and CPU requirements.

Type

Type defines the type of filter. Note that different filters may consume different amounts of CPU. By definition a filter does not produce any frequencies which are not already in the signal, hence the name "filter". The difference between the types is how each filter modifies the levels of each frequency. Some filters completely remove certain frequencies, others just change the levels of certain frequencies. If you wish to make the signal richer by creating additional frequencies which are NOT in the signal yet, use a distortion or saturation plugin.

Low-pass, high-pass, band-pass and notch filter out some frequencies completely. Low-pass filter, for example, lets all frequencies below a certain limit pass and removes everything above. This is possible only in theory though, so you might say that the higher the frequency is above the filter frequency, the more it is attenuated. The higher the slope is, the steeper the filter is, hence it removes more of the unwanted frequencies. Traditional low-pass filters have a 12dB/octave slope, which means that, for example, if you have that filter set at 1kHz and the Q is configured so that at 1kHz the gain is -3dB (which is usually the default, technical reasons), then at 2kHz (+1 octave) it is -15dB, at 4kHz (+2 octaves) it is -27dB etc. Our filters can provide up to 120dB/octave slope, so it can pretty much kill everything above it within a single octave.
High-pass filter works the same way, but kills everything below its frequency. Notch kills everything at the filter frequency plus some adjacent frequency range (determined by the filter's Q value), while band-pass works the other way around - it only lets through the filter frequency and the adjacent frequency range.

Peak and shelf filters are similar to those used in equalizers.

Fade filters provide cross-fades between low-pass and high-pass filters and other combinations. Use the **Character** parameter to control how much LP and how much HP is used then.

Harmonics filters are complex combinations of peak filters designed to process multiple harmonics of the base frequency. Basically if you configure a harmonic filter at say 100Hz, then there will be series of peak filters at 100Hz, 200Hz, 400Hz etc. or (100Hz, 200Hz, 300hz... if the linear version is used). The **character** parameter controls the level of succeeding harmonics. For example, if character is 0%, then it is basically just an ordinary peak filter. If it is 100%, then there is one filter for all available harmonics, each with the same gain. For something in between, the gain for each higher harmonic is lower than the previous one.
Linear harmonics filters affect linear multiples of the base frequency, while normal harmonics filters only affect power-2 multiples, hence octaves above the base. **Swap** versions cause inverted gain for odd and even harmonics.

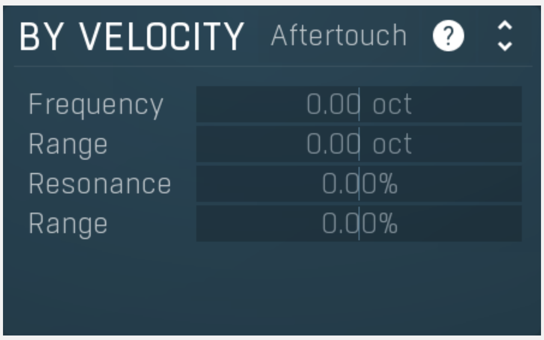
Sub-X, over-X and band-X filters are specialized complex combinations of other filters originally designed for wobbling basses. These mainly combine LP/HP/BP filters with harmonic filters. The **character** parameter controls the distribution of harmonics and should be used simply by trial-and-error.

Formant filters are filters emphasizing vowel sounds. There are filters for each vowel and the newest filter, called **Formant A-E-I-O-U** cross-fades between these 5 vowels, depending on the **character** parameter. To get reasonable "talkbox" sounds, it is recommended to use a rich audio signal (e.g. saw wave).

Comb and diffuser filters are complex comb filtering processors with pretty wild and fat responses. These range from simple comb filtering to complex almost ambient responses. Each filter uses a different kernel, so it shall be selected by trial-and-error approach. The**character** parameter controls the internal feedback of the filter.

Polymorph filters are generic polymorphic filters, which change its internal structure according to the **Character** parameter and provide a virtually limitless number of unique sound combinations. However, these are usually also the most computationally demanding.

By velocity panel



By velocity panel lets you control how much the basic parameters are affected by the note velocity. Unlike modulators and MIDI, which are global and affect all voices when changed, these parameters lets you assign different values to each voice depending on the velocity (or aftertouch).

Aftertouch

Aftertouch button enables processing aftertouch for this filter in the same way as velocity.

Frequency by velocity

Frequency by velocity controls how much the filter frequency is adjusted depending on the velocity of each note. For example, if this

is set to +1 octave and the filter **frequency parameter** is +2 octaves, then for a note with minimum velocity the filter frequency will be +2 octaves. If the note has maximum velocity, then the filter frequency will be +3 octaves.
Range: -8.00 oct to +8.00 oct, default 0.00 oct

Frequency range by velocity

Frequency range by velocity controls how much the filter frequency range is adjusted depending on the velocity of each note. For example, if this is set to +1 octave and filter **frequency range parameter** is +2 octaves, then for a note with minimum velocity the filter frequency range will be +2 octaves. If the note has maximum velocity, then the filter frequency range will be +3 octaves.
Range: -8.00 oct to +8.00 oct, default 0.00 oct

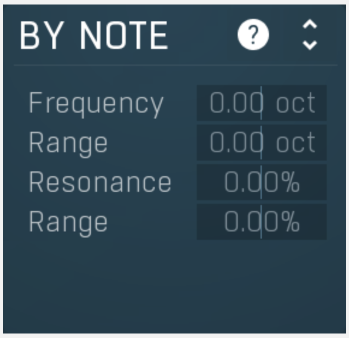
Resonance by velocity

Resonance by velocity controls how much the filter central resonance is adjusted depending on the velocity of each note. For example, if this is set to +30% and filter **resonance parameter** is 40%, then for a note with minimum velocity the filter resonance will be 40% of an octave. If the note has maximum velocity, then the filter resonance will be 70%.
Range: -100.0% to 100.0%, default 0.00%

Resonance range by velocity

Resonance range by velocity controls how much the filter resonance range is adjusted depending on the velocity of each note. For example, if this is set to +30% and filter **resonance range parameter** is 40%, then for a note with minimum velocity the filter resonance range will be 40% of an octave. If the note has maximum velocity, then the filter resonance range will be 70%.
Range: -100.0% to 100.0%, default 0.00%

By note panel



By note panel lets you control how much the basic parameters are affected by which note is being pressed. Unlike modulators and MIDI, which are global and affect all voices when changed, these parameters let you assign different values to each voice depending on the note.

Frequency by note

Frequency by note controls how much the filter frequency is adjusted depending on which note is being pressed. For example, if this is +1 octave and filter **frequency parameter** is +2 octaves, then for the lowest note the filter frequency will be +2 octaves. For the highest note the filter frequency will be +3 octaves.
Range: -8.00 oct to +8.00 oct, default 0.00 oct

Frequency range by note

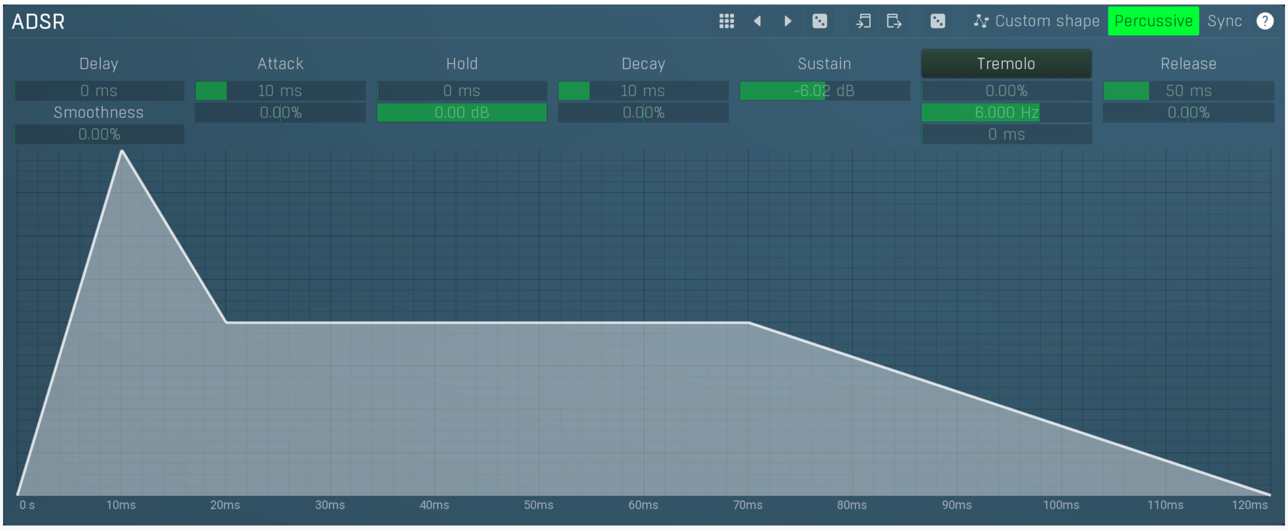
Frequency range by note controls how much the filter frequency range is adjusted depending on which note is being pressed. For example, if this is +1 octave and filter **frequency range parameter** is +2 octaves, then for the lowest note the filter frequency range will be +2 octaves. For the highest note the filter frequency range will be +3 octaves.
Range: -8.00 oct to +8.00 oct, default 0.00 oct

Resonance by note

Resonance by note controls how much the filter central resonance is adjusted depending on which note is being pressed. For example, if this is +30% and filter **resonance parameter** is 40%, then for the lowest note the filter resonance will be 40% of an octave. For the highest note the filter resonance will be 70%.
Range: -100.0% to 100.0%, default 0.00%

Resonance range by note

Resonance range by note controls how much the filter resonance range is adjusted depending on which note is being pressed. For example, if this is +30% and filter **resonance range parameter** is 40%, then for the lowest note the filter resonance range will be 40% of an octave. For the highest note the filter resonance range will be 70%.
Range: -100.0% to 100.0%, default 0.00%



ADSR graph

ADSR graph controls the ADSR envelope for the filter.Our ADSR envelopes are much more sophisticated than classic attack-decay-sustain-release envelopes. Besides these common parameters they also let you control the curvature of each stage. Additionally, there are hold and delay sections ("DAHDSR"), global smoothing and tremolo. You can even use the custom shape mode to define your own attack/release curves.

Presets

Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.

Left arrow

Left arrow button loads the previous preset.

Right arrow

Right arrow button loads the next preset.

Randomize

Randomize button loads a random preset.

Copy

Copy button copies the settings onto the system clipboard.

Paste

Paste button loads the settings from the system clipboard.

Random

Random button generates random settings using the existing presets.

Custom shape

Custom shape button enables custom shape mode, which lets you draw your own attack and release stages using the envelope system. Both stages are then automatically connected to form the resulting envelope.

Percussive

Percussive button activates the immediate release mode in which case the note-off causes an immediate switch to the release stage. If this is disabled, the release stage does not occur until the whole attack/decay stage finishes.

Sync

Sync button controls the ADSR tempo sync feature. By default this is disabled and means that all times are followed exactly, meaning that if **Attack** is say 100ms, then it will be 100ms indeed. Tempo sync lets the plugin adjust the times to ensure it will be always in sync with the host tempo. In this case 100ms may become say 125ms if the tempo is 120bpm, because 125ms is the length of a 16th note. This makes it extremely simple to convert any envelope to a tempo-synced one. The plugin always chooses the nearest longer note, in other words it always round up.
Straight and **Triplets** modes automatically find 'nice' values.
For example, if a 16th note takes 100ms, the attack time is 550ms, and the sync mode is straight, then the plugin checks for 100ms, find out that it is too low, so it checks 8th note, being 200ms, still too low, then continues with quarter note, which takes 400ms, and still not enough, finally 800ms corresponding to a half note is the one, so the resulting time will be 800ms. Triplet cases are more complex, but the principle is the same.

1/16, **1/8** and **1/4** modes choose the nearest higher multiply of the base note length. For example, if a 16th note takes 100ms, the attack time is 550ms, and the sync mode is 1/16, the resulting time will be 600ms.

Tremolo

Tremolo

Tremolo button displays additional tremolo settings, containing tremolo behaviour and shape.

0 ms

Delay

Delay lets you shift the entire envelope forwards in time. While this doesn't make much sense for a global instrument envelope for instance, it may be well useful to control characteristics of evolving sounds.

10 ms

Attack

Attack controls the length of the initial stage of the envelope. It is one of the most important parameters controlling how quick the initial transient is. For most instruments the length is quick short, but for pads and other slowly evolving sounds it is quite common to set this to several seconds.

0 ms

Hold

Hold specifies the time the level stays at maximum after the attack stage.

10 ms

Decay

Decay controls the time it takes for the level to drop from the maximum to the **Sustain**. If the sustain is 0dB, then this parameter has no effect, because in a way the sustain stage starts immediately after the attack.

-6.02 dB

Sustain

Sustain controls the sustain level. For most sounds the initial attack transient is the highest point of the entire sound. Imagine playing a string instrument, such as a guitar, the initial hit to the strings is represented by the attack+hold+decay sections and is the most prominent. After that the level drops to the sustain stage, where it holds for most of the time.

0.00%

Tremolo

Tremolo defines the amount of the tremolo effect that is engaged in the sustain, or even in the decay section and continues until the envelope ends. While this is a rather unusual feature for an envelope to have, it is very handy for simulating various effects human players do when performing on real instruments, such as the tremolo or vibrato.

50 ms

Release

Release controls the length of the release section, which usually starts when a note is released.

0.00%

Attack shape

Attack shape controls the shape of the attack section and defines its sound character.

0.00 dB

Hold level

Hold level controls the level of the hold section. By default it equals maximum meaning that the hold section actually holds the maximum level. However by making it lower you can sort of simulate 2 separate decay sections, first going from maximum to hold level, second going from hold level to sustain.

0.00%

Decay shape

Decay shape controls the shape of the decay section and defines its sound character.

6.000 Hz

Tremolo rate

Tremolo rate controls the speed of the tremolo. In the tremolo settings it is possible to control additional characteristics including tempo sync.

0.00%

Release shape

Release shape controls the shape of the release section and defines its sound character.

0.00%

Smoothing

Smoothing lets you smoothen the entire envelope avoiding abrupt jumps. Note that in some cases involving short jumps the results may be a bit obscure.

0 ms

Tremolo fade-in

Tremolo fade-in defines the time for the tremolo to reach its full level. It is a natural behaviour of human players (on say a saxophone) that they don't start a full tremolo immediately and rather let the modulation rise to maximum over a period of time.

FX panel

FX panel controls the effects engine. The effects are applied to all voices at once, so for example distortion in the filters reacts very differently from distortion in the effects.
The modular pipeline makes dozens of ultra-high quality effects available and you can connect them together in any way that you want. The signal always flows from top to bottom, left to right through various processors. The engine can also generate complete structures, just in case you need some inspiration. The top row contains the available inputs and the bottom row shows the outputs, which are simply mixed together. You can disable any output by clicking on it (that also disconnects the processors in that chain).
Click on an empty field to add a processor to it. Use **right mouse button** (or **Ctrl + click**) on a processor to get some additional settings, mostly about routing audio and MIDI. There are also a few shortcuts: **double click** enables or disables the processor, **Alt + click** deletes the processor, and **Shift + click** displays the processor in a popup window.

The set of processors includes everything from modulation effects, delays, distortions, dynamics, filters etc., and also contains several building blocks - mixer, crossover, LR <-> MS (de)encoder, LFO, channel matrix and many more. Each processor has some inputs and outputs, in most cases it is just 1 input and 1 output as you might expect. There are also processors with side-chain inputs (e.g. Ratio or Dynamics), processors with multiple outputs (e.g. Crossover), processors with multiple inputs (e.g. Mixer) and even processors with no inputs (e.g. Oscillator, LFO). The inputs and outputs are intelligently mapped by default, in most cases the whole structure "connects", but sometimes you may need to use the **right mouse button** to get the additional routing settings.

Most processors are self-explanatory, but a few of them could use some additional information:

Modular processor is actually "itself", the whole modular pipeline inside itself. This is useful for example when the structure is starting to get cluttered. Note that you may run into GUI problems - it is impossible to fit that amount of controls onto a screen without huge resolution, so you may need to use the **shift + click** to get the modular processor displayed in a pop-up window.

Feedback processor serves as a feedback source. Normally the feedback would not be possible, because the signal always flows from top to bottom. The feedback processor lets you go around this. It has no inputs, but generates a signal anyway - by taking it from somewhere else, from previous processing to be precise. Any of the processors can "generate feedback" into one of the feedback channels. You can imagine that each processor can not only process the signal somehow and pass the results to the output, it can also store the output in one of the feedback channels, just in case some feedback processor would use it.

The feedback processor does not have many parameters, but 2 of them are very important - feedback and delay. First, feedback is the input gain. If it is too high, reaching 0dB, you may expect the typical infinite feedback screaming. The processor provides a clipper, but it will not save you, so be careful when using it. Second, delay - it is physically impossible to have zero delay feedback, not in the analog world and not in the digital world either. So the delay controls how much time it takes to get back. And here is the catch - for higher delays, say above 5ms, there are no problems, but with lower delays the CPU consumption can rise exponentially. There are technical reasons for that and it can never be improved, think about it as physical limitations.
How to use the feedback then? First, create a feedback processor somewhere, it will use the feedback channel 1 by default. Then right-click on any of the existing processors and set "generate feedback 1", that is all. If you do not have any processors there and you just want to use feedback as some kind of experimental delay generator, use some low-CPU processor, such as Utility.

Presets

Presets button shows a window with all available presets. A preset can be loaded from the preset window by double-clicking on it, selecting via the buttons or by using your keyboard. You can also manage the directory structure, store new presets, replace existing ones etc. Presets are global, so a preset saved from one project, can easily be used in another. The arrow buttons next to the preset button can be used to switch between presets easily.

Holding **Ctrl** while pressing the button loads a random preset. There must be some presets for this feature to work of course.

Presets can be backed up by 3 different methods:
A) Using "Backup" and "Restore" buttons in each preset window, which produces a single archive of all presets on the computer.
B) Using "Export/Import" buttons, which export a single folder of presets for one plugin.
C) By saving the actual preset files, which are found in the following directories (not recommended):
Windows: C:\Users\{username}\AppData\Roaming\MeldaProduction
Mac OS X: /Library/Application support/MeldaProduction

Files are named based on the name of the plugin like this: "{pluginname}.presets", so for example MAutopan.presets or MDynamics.presets. If the directory cannot be found on your computer for some reason, you can just search for the particular file.

Please note that prior to version 16 a different format was used and the naming was "{pluginname}presets.xml". *The plugin also supports an online preset exchange. If the computer is connected to the internet, the plugin connects to our server once a week, submits your presets and downloads new ones if available. This feature is manually maintained in order to remove generally unusable presets, so it may take some time before any submitted presets become available. This feature relies on each user so we strongly advise that any submitted presets be named and organised in the same way as the factory presets, otherwise they will be removed.*

Left arrow
Left arrow button loads the previous preset.

Right arrow
Right arrow button loads the next preset.

Randomize
Randomize button loads a random preset.

Copy
Copy button copies the settings onto the system clipboard.

Paste
Paste button loads the settings from the system clipboard.

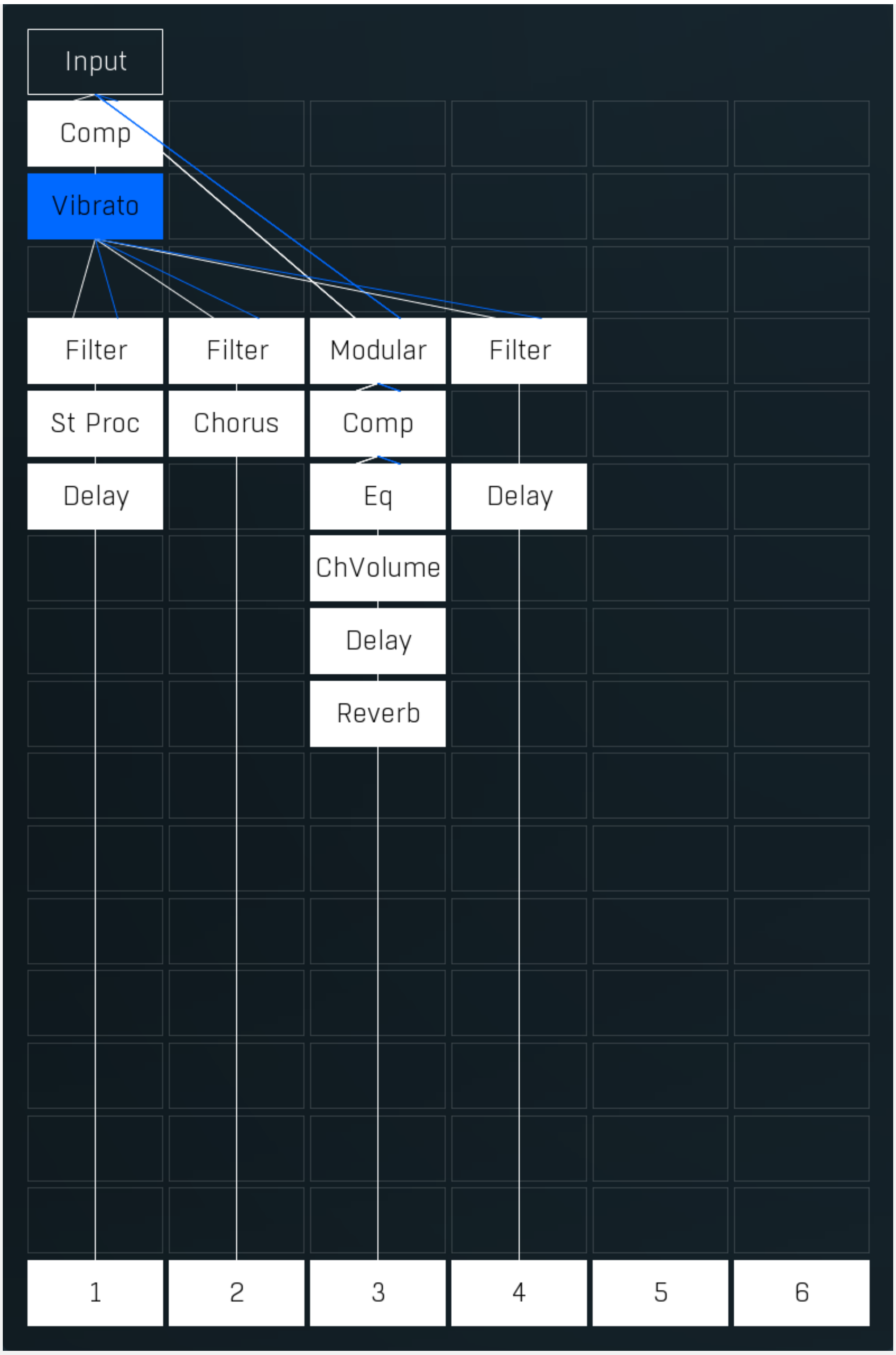
Randomize
Randomize button (with the text 'Random') generates random settings. Generally, randomization in plug-ins works by selecting random values for all parameters, but rarely achieves satisfactory results, as the more parameters that change the more likely one will cause an unwanted effect. Our plugins employ a smart randomization engine that learns which settings are suitable for randomization (using the existing presets) and so is much more likely to create successful changes.

In addition, there are some mouse modifiers that assist this process. The smart randomization engine is used by default if no modifier keys are held.

Holding **Ctrl** while clicking the button constrains the randomization engine so that parameters are only modified slightly rather than completely randomized. This is suitable to create small variations of existing interesting settings.

Holding **Alt** while clicking the button will force the engine to use full randomization, which sets random values for all reasonable automatable parameters. This can often result in "extreme" settings. Please note that some parameters cannot be randomized this way.

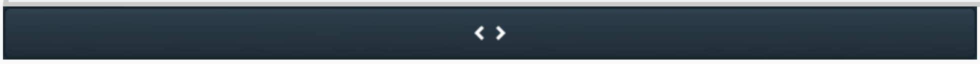
Modular editor



Modular editor lets you edit the processing flow. There is at least one input at the top, several outputs at the bottom and several processing boxes in-between. Processing is always performed from top to bottom and is indicated by a solid line. All outputs except the first one are disabled by default, unless you put a plugin into its chain. Every output can be disabled by clicking on it.

Each processing box can contain one plugin. By default all plugins are taking their inputs directly from above (thus from the same chain that they are in), but if you for example put a plugin in the first row of the second chain, there is no input above, so it is redirected to the first input. Some plugins may have multiple input channels or even a variable number of input channels. Most plugins have just one output channel, in which case they place the output into the same chain that they are in. Some plugins have variable output channels in which case they send them to the same channel, the next one to the right and so on.

If you **click** on an empty box a menu with all available processing plugins will be shown. If you click on a box which already contains a plugin, it will be selected and its editor shown to the right. The currently-selected plugin is highlighted.
If you hold **Ctrl** and click on the plugin or click using **right mouse button**, the advanced settings window will be displayed. You can use it to configure the plugin input and output channels and much more.
Double-click on the plugin to enable or disable it.
Hold **Alt** and click on the plugin to delete it.
Hold **Shift** and click on the plugin to show its editor in a pop-up window.



Collapse

Collapse button minimizes or enlarges the panel to release space for other editors.

Vibrato 1

Presets

Presets button shows a window with all available presets. A preset can be loaded from the preset window by double-clicking on it, selecting via the buttons or by using your keyboard. You can also manage the directory structure, store new presets, replace existing ones etc. Presets are global, so a preset saved from one project, can easily be used in another. The arrow buttons next to the preset button can be used to switch between presets easily.

Holding **Ctrl** while pressing the button loads a random preset. There must be some presets for this feature to work of course.

Presets can be backed up by 3 different methods:
A) Using "Backup" and "Restore" buttons in each preset window, which produces a single archive of all presets on the computer.
B) Using "Export/Import" buttons, which export a single folder of presets for one plugin.
C) By saving the actual preset files, which are found in the following directories (not recommended):
Windows: C:\Users\{username}\AppData\Roaming\MeldaProduction
Mac OS X: /Library/Application support/MeldaProduction

Files are named based on the name of the plugin like this: "{pluginname}.presets", so for example MAutopan.presets or MDynamics.presets. If the directory cannot be found on your computer for some reason, you can just search for the particular file.

Please note that prior to version 16 a different format was used and the naming was "{pluginname}presets.xml". *The plugin also supports an online preset exchange. If the computer is connected to the internet, the plugin connects to our server once a week, submits your presets and downloads new ones if available. This feature is manually maintained in order to remove generally* *unusable presets, so it may take some time before any submitted presets become available. This feature relies on each user so we strongly advise that any submitted presets be named and organised in the same way as the factory presets, otherwise they will be removed.*

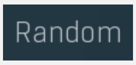
Left arrow
Left arrow button loads the previous preset.

Right arrow
Right arrow button loads the next preset.

Randomize
Randomize button loads a random preset.

Copy
Copy button copies the settings onto the system clipboard.

Paste
Paste button loads the settings from the system clipboard.



Randomize
Randomize button (with the text 'Random') generates random settings. Generally, randomization in plug-ins works by selecting random values for all parameters, but rarely achieves satisfactory results, as the more parameters that change the more likely one will cause an unwanted effect. Our plugins employ a smart randomization engine that learns which settings are suitable for randomization (using the existing presets) and so is much more likely to create successful changes.

In addition, there are some mouse modifiers that assist this process. The smart randomization engine is used by default if no modifier keys are held.

Holding **Ctrl** while clicking the button constrains the randomization engine so that parameters are only modified slightly rather than completely randomized. This is suitable to create small variations of existing interesting settings.

Holding **Alt** while clicking the button will force the engine to use full randomization, which sets random values for all reasonable automatable parameters. This can often result in "extreme" settings. Please note that some parameters cannot be randomized this way.

Channel mode
Channel mode button shows the current processing channel mode, e.g. **Left+Right (L+R)** indicates the processing of left and right channels. This is the default mode for mono and stereo audio material and effectively processes the incoming signal as expected. However the plugin also provides additional modes, of which you may take advantage as described below. Mastering this feature will give you unbelievable options for controlling the stereo field.

Note that this is not relevant for mono audio tracks, because the host supplies only one input and output channel.

Left (L) mode and Right (R) mode allow the plugin to process just one channel, only the left or only the right. This feature has a number of simple uses. Equalizing only one channel allows you to fix spectral inconsistencies, when mids are lower in one channel for example. A kind of stereo expander can be produced by equalizing each side differently. Stereo expansion could also be produced by using a modulation effect, such as a vibrato or flanger, on one of these channels. Note however that the results would not be fully mono compatible.

Left and right channels can be processed separately with different settings, by creating two instances of the plugin in series, one set to 'L' mode and the other to 'R' mode. The instance in 'L' mode will not touch the right channel and vice versa. This approach is perfectly safe and is even advantageous, as both sides can be configured completely independently with both settings visible next to each other.

Mid (M) mode allows the plugin to process the so-called mid (or mono) signal. Any stereo signal can be transformed from left and right, to mid and side, and back again, with minimal CPU usage and no loss of audio quality. The mid channel contains the mono sum (or centre), which is the signal present in both left and right channels (in phase). The side channel contains the difference between the left and right channels, which is the "stereo" part. In 'M mode' the plugin performs the conversion into mid and side channels, processes mid, leaves side intact and converts the results back into the left and right channels expected by the host.

To understand what a mid signal is, consider using a simple gain feature, available in many plugins. Setting the plugin to M mode and decreasing gain, will actually lower or attenuate the mono content and the signal will appear "wider". There must be some stereo content present, this will not work for monophonic audio material placed in stereo tracks of course. Similarly amplifying the mono content by increasing the gain, will make the mono content dominant and the stereo image will become "narrower".

As well as a simple gain control there are various creative uses for this channel mode.
Using a **compressor** on the mid channel can widen the stereo image, because in louder parts the mid part gets attenuated and the stereo becomes more prominent. This is a good trick to make the listener focus on an instrument whenever it is louder, because a wider stereo image makes the listener feel that the origin of the sound is closer to, or even around them.

A **reverb** on the mid part makes the room appear thin and distant. It is a good way to make the track wide due to the existing stereo content, yet spacey and centered at the same time. Note that since this effect does not occur naturally, the result may sound artificial on its own, however it may help you fit a dominant track into a mix.
An **equalizer** gives many possibilities - for example, the removal of frequencies that are colliding with those on another track. By processing only the mid channel you can keep the problematic frequencies in the stereo channel. This way it is possible to actually fit both tracks into the same part of the spectrum - one occupying the mid (centre) part of the signal, physically appearing further away from the listener, the other occupying the side part of the signal, appearing closer to the listener.
Using various **modulation effects** can vary the mid signal, to make the stereo signal less correlated. This creates a wider stereo image and makes the audio appear closer to the listener.

Side (S) mode is complementary to M mode, and allows processing of only the side (stereo) part of the signal leaving the mid intact. The same techniques as described for M mode can also be applied here, giving the opposite results.
Using a **gain** control with positive gain will increase the width of the stereo image.
A **compressor** can attenuate the side part in louder sections making it more monophonic and centered, placing the origin a little further away and in front of the listener.
A **reverb** may extend the stereo width and provide some natural space without affecting the mid content. This creates an interesting side-effect - the reverb gets completely cancelled out when played on a monophonic device (on a mono radio for example). With stereo processing you have much more space to place different sounds in the mix. However when the audio is played on a monophonic system it becomes too crowded, because what was originally in two channels is now in just one and mono has a very limited capability for 2D placement. Therefore getting rid of the reverb in mono may be advantageous, because it frees some space for other instruments.
An **equalizer** can amplify some frequencies in the stereo content making them more apparent and since they psycho acoustically become closer to the listener, the listener will be focused on them. Conversely, frequencies can be removed to free space for other instruments in stereo.
A **saturator / exciter** may make the stereo richer and more appealing by creating higher harmonics without affecting the mid channel, which could otherwise become crowded.
Modulation effects can achieve the same results as in mid mode, but this will vary a lot depending on the effect and the audio material. It can be used in a wide variety of creative ways.

Mid+Side (M+S) lets the plugin process both mid and side channels together using the same settings. In many cases there is no difference to L+R mode, but there are exceptions.
A **reverb** applied in M+S mode will result in minimal changes to the width of the stereo field (unless it is true-stereo, in which case mid will affect side and vice versa), it can be used therefore, to add depth without altering the width.
A **compressor** in M+S mode can be a little harder to understand. It basically stabilizes the levels of the mid and side channels. When channel linking is disabled in the compressor, you can expect some variations in the sound field, because the compressor will attenuate the louder channel (usually the mid), changing the stereo width depending on the audio level. When channel linking is enabled, a compressor will usually react similarly to the L+R channel mode.
Exciters or saturators are both nonlinear processors, their outputs depend on the level of the input, so the dominant channel (usually mid) will be saturated more. This will usually make the stereo image slightly thinner and can be used as a creative effect.

How to modify mid and side with different settings? The answer is the same as for the L and R channels. Use two instances of the plugin one after another, one in M mode, the other in S mode. The instance in M mode will not change the side channel and vice versa.

Left+Right(neg) (L+R-) mode is the same as L+R mode, but the the right channel's phase will be inverted. This may come in handy if the L and R channels seem out of phase. When used on a normal track, it will force the channels out of phase. This may sound like an extreme stereo expansion, but is usually extremely fatiguing on the ears. It is also not mono compatible - on a mono device the track will probably become almost silent. Therefore be advised to use this only if the channels are actually out of phase or if you have some creative intent.

There are also 4 subsidiary modes: **Left & zero Right (L(R0))**, **Right & zero Left (R(L0))**, **Mid & zero Side (M(S0))** and **Side & zero Mid (S(M0))**. Each of these processes one channel and silences the other.

Surround mode is not related to stereo processing but lets the plugin process up to 8 channels, depending on how many the host supplies. For VST2 plugins you have to first activate surround processing using the **Activate surround** item in the bottom. This is a global switch for all MeldaProduction plugins, which configures them to report 8in-8out capabilities to the host, on loading. It is disabled by default, because some hosts have trouble dealing with such plugins. After activation, restart your host to start using the surround capabilities of the plugins. Deactivation is done in the same way. Please note that all input and output busses will be multi-channel, that includes side-chain for example. For VST3/AU/AAX plugins the activation is not necessary.

First place the plugin on a surround track - a track that has more than 2 channels. Then select **Surround** from the plug-in's Channel Mode menu. The plugins will regard this mode as a natural extension of 2 channel processing. For example, a compressor will process each channel separately or measure the level by combining the levels of all of the inputs provided. Further surround processing properties, to enable/disable each channel or adjust its level, can be accessed via the **Surround settings** in the menu.

Ambisonics mode provides support for the modern 3D systems (mostly cinema and VR) with up to 64 channels (ambisonics 7th order). Support for this is still quite rare among the DAWs, so this needs to be activated in all DAWs using the **Activate ambisonics** item in the bottom. This is a global switch for all MeldaProduction plugins, which configures them to report 64in-64out capabilities to the host, on loading. After activation, restart your host to start using the ambisonics capabilities of the plugins. Deactivation is done in the same way. Please note that all input and output busses will be multi-channel, that includes side-chain for example.

First place the plugin on an ambisonics track, supported are all orders from 1st (4 channels) to 7th (64 channels). Then select **Ambisonics** from the plug-in's Channel Mode menu. Finally select the **Ambisonics settings** in the menu and configure the Ambisonics order and other settings if needed. The plugins will regard this mode as a natural extension of 2 channel processing. For example, a compressor will process each channel separately or measure the level by combining the levels of all of the inputs provided.

Oversampling

Oversampling can potentially improve sound quality by processing at a higher sample rate. Processors such as compressors, saturators, distortions etc., which employ nonlinear processing generate higher harmonics of the existing frequencies. If these frequencies exceed the Nyquist rate, which equals half of the sampling rate, they get mirrored back under the Nyquist rate. This is known as aliasing and is almost always considered an artifact. This is because the mirrored frequencies are no longer harmonic and sound as digital noise as this effect does not physically occur in nature. Oversampling reduces the problem by temporarily increasing the sampling rate. This moves the Nyquist frequency which in turn, diminishes the level of the aliased harmonics. Note that the point of oversampling is not to remove harmonics, we usually add them intentionally to make the signal richer, but to reduce or attenuate the harmonics with frequencies so high, that they just cannot be represented within the sampling rate.

To understand aliasing, try this experiment: Set the sampling rate in your host to 44100 Hz. Open MOscillator and select a "rectangle" or "full saw" waveform. These simple waveforms have lots of harmonics and without oversampling even they become highly aliased. Now select 16x oversampling and listen to the difference. If you again select 1x oversampling, you can hear that the audio signal gets extensively "dirty". If you use an analyzer (MAnalyzer or MEqualizer for example), you will clearly see how, without oversampling, the plugin generates lots of inharmonic frequencies, some of them which are even below the fundamental frequency. Here is another, very extreme example to demonstrate the result of aliasing. Choose a "sine" shape and activate 16x oversampling. Now use a distortion or some saturation to process the signal. It is very probable that you will be able to hear (or at least see in the analyzer) the aliased frequencies.

The plugin implements a high-quality oversampling algorithm, which essentially works like this: First the audio material is upsampled to a higher sampling rate using a very complicated filter. It is then processed by the plugin. Further filtering is performed in order to remove any frequencies above the Nyquist rate to prevent aliasing from occurring, and then the audio gets downsampled to the original sampling rate.

Oversampling also has several disadvantages of which you should be aware before you start using it. Firstly, upsampled processing induces latency (at least in high-quality mode, although you can select low-quality directly in this popup), which is not very usable in real time applications. Secondly, oversampling also takes much more CPU power, due to both the processing being performed at a higher sampling rate (for 16x oversampling at 44100 Hz, this equates to 706 kHz!), and the complex filtering. Finally, and most importantly, oversampling creates some artifacts of its own and for some algorithms processing at higher sampling rates can actually lower the audio quality, or at least change the sound character. Your ears should always be the final judge.

As always, use this feature ONLY if you can actually hear the difference. It is a common misconception that oversampling is a miraculous cure all that makes your audio sound better. That is absolutely not the case. Ideally, you should work in a higher sampling rate (96kHz is almost always enough), while limiting the use of oversampling to some heavily distorting processors.

Show window

Show window button displays the plugin in a dedicated pop-up window. You can do the same thing by Shift + click on the processor item in the modular grid.

Depth

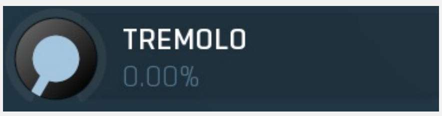
Depth defines how powerful the effect is.
Range: 0 ms to 10 ms, default 1.0 ms

Rate

Rate defines the speed of the low frequency oscillator. This is available only when synchronization to host is disabled.
Range: 0.0100 Hz to 100.0 Hz, default 4.000 Hz

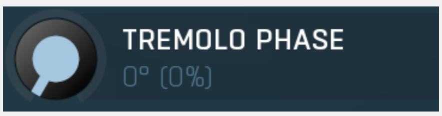
Width

Width defines the phase difference between particular channels. This is very simple and often practical way to accomplish a kind of stereo expansion.
Range: -360Â° (-100.0%) to 360Â° (100.0%), default 0Â° (0%)



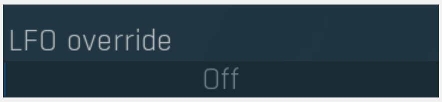
Tremolo

Tremolo defines how powerful the additional tremolo effect is.
Range: 0.00% to 100.0%, default 0.00%



Tremolo phase

Tremolo phase defines the phase offset of the tremolo generator to the vibrato shape generator.
Range: 0Â° (0%) to 360Â° (100.0%), default 0Â° (0%)

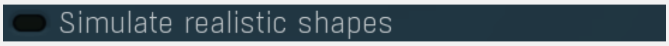


LFO override

LFO override lets you override the LFO and control the modulation value directly. This feature may offer several creative possibilities. You can then either automate it or even better, use the modulators (if the plugin provides any) to follow the input level, pitch, randomize etc. Set this below -1 to disable this feature. Please note that since there is only one parameter, by using it you will lose the possibility of having different values for each channel, hence potential stereoizing capabilities will not be available.
Range: Off to 100.0%, default Off

Invert tremolo phase

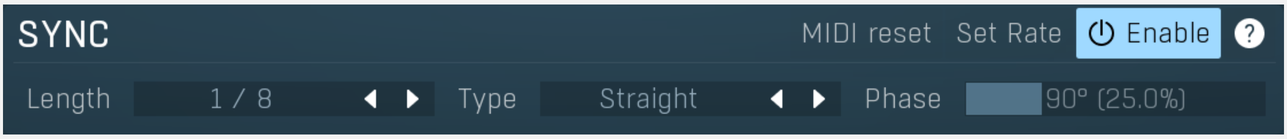
Invert tremolo phase inverts the resulting tremolo shape.



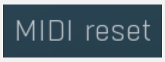
Simulate realistic shapes

Simulate realistic shapes enables advanced processing which tries to approximate the response of real vibrato. This is unfortunately not possible with digital signals, only synthesizers can do that when generating the sound.

Synchronization panel



Synchronization panel contains parameters for the to-host synchronization.



MIDI reset

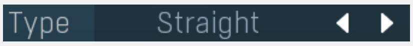
MIDI reset button displays the settings for the MIDI reset feature, which can reset the LFO based on incoming MIDI notes.

Set Rate

Set Rate button sets the **Rate** parameter used when sync is disabled according to current sync speed. This is useful when you want to leave the oscillator unsynchronized, however you want to start with the current synced speed.

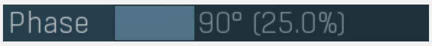
Length

Length defines the note length to be used.



Type

Type defines the note type, such as straight notes or triplets, to be used. Together the **Length** and **Type** determine the actual time/delay.
Example: '1/4 Straight' at 120 bpm = a delay of 500 ms, '1/4 Triplet' at 160 bpm = a delay of 281.25 ms.



Phase

Phase defines the phase offset of the to-host synchronization.
Range: 0Â° (0%) to 360Â° (100.0%), default 90Â° (25.0%)

Signal graph

Signal graph defines the low frequency oscillator shape.Signal-generator is an incredibly versatile generator of low & high frequency signals. It offers 2 distinct modes - Normal and Harmonics.
Normal mode is appropriate for low-frequency oscillators, where the graphical shape is relevant and is used to drive some form of modulation. For example, a tremolo uses this modulation to change the actual signal level in time. Frequencies for such oscillators usually do not exceed 20Hz as this is a sort of limit above which the frequencies become audible.
Harmonics mode is designed for high-frequency oscillators, where the actual shape is not as important as the harmonic content of the resulting signal, hence it is especially useful for actual audio signals. Please note that since a shape can contain more harmonics than those available from the harmonic generator, the results may not be exactly the same. As an example, a rectangular wave in normal mode may sound fuller than when converted to the harmonic mode.

Use the arrow-down button to switch from normal mode to harmonics mode or click the **Normal** and **Harmonics** buttons

Normal mode

The generator first uses a set of predefined signal shapes (sine, triangle, rectangle...), which you can select directly by right-clicking on the editor and choosing the requested shape from the menu. This menu also provides a link to the modulator shapes preset manager, normalization and randomization. You can also use the **Main shape** parameter, which generates a combination of adjacent signals to provide a nearly inexhaustible number of basic shapes.

The engine then combines the predefined shape with a **Custom shape**, which may be anything you can draw using the advanced envelope engine, depending on the level set by the **Custom shape** control. Use the **Edit** button to edit the custom shape.

You can also combine those results with a fully featured step sequencer, with variable number of steps and several shapes for each of them, depending on the level set by the **Step sequencer** control. Use the lower **Edit** button to edit the step sequence.

Those results may be mixed with a custom sample, which is available from the advanced settings, accessed by clicking the **Advanced** button.

Smoothness softens any abrupt edges, generated by the step sequencer for example.

Finally there are **Advanced** features providing more complex transformations, adding harmonics etc. or you can click the **Randomize** button in the top-left corner to generate a random, but reasonable, modulator shape.

Harmonics mode

Harmonics mode represents the signal as a series of harmonics (that is, multiples of the base frequency). For example, when your oscillator has a frequency of 2Hz (set in the **Rate** panel), then the harmonics are 2Hz, 4Hz, 6Hz, 8Hz etc. In theory, any signal can be created by mixing a potentially infinite number of these harmonics.

The harmonics mode lets you control the levels and phases of each harmonic. The top graph controls the levels of individual harmonics, while the bottom one controls their phases. Use the left-mouse button to change the values in each graph, the right-mouse button sets the default for the harmonics - 0% level and 0% phase. In both graphs the harmonics of power 2 (that is octaves) are highlighted. Other harmonics may actually sound disharmonic, despite their names.

For example, if you reset all harmonics to the defaults and increase only the first one, you will get a simple sine wave. By adding further harmonics you make the output signal more complex.

Harmonics controls the number of generated harmonics. The higher the number is, the richer the output signal is (unless the levels are 0% of course). This is useful to make the sound cleaner. For example, if you transform a saw-tooth wave to harmonics, it would not sound like a typical saw-tooth wave anymore, but more like a low-passed version of one. The more harmonics you use, the closer you get to the original saw-tooth wave.

Generator is a powerful tool for generating the harmonics, which are otherwise rather clumsy to edit. The generator provides several parameters based upon which it creates the entire series of harmonic levels and phases. These parameters are usually easier to understand than the harmonics themselves. Part of the generator is the randomizer available via the **Random seed** button, which smartly generates random settings for the generator. This makes the process of getting new sounds as simple as possible.

Signal generation fundamentals

The signal generator produces a periodic signal with specified wave shape. This means that the signal is repeating over and over again. As a result it can only contain multiples of the fundamental frequency. For example, if the generator is producing 100Hz signal, then it can contain 100Hz (fundamental or 1st harmonic), 200Hz (2nd harmonic), 300Hz (3rd harmonic), 400Hz (4th harmonic) etc. However, it can never produce 110Hz. You can then control the level of each harmonic and their relative phases. It does not matter whether you use the normal mode using oscillator shapes, or harmonics mode where you can control the harmonics directly. If both modes result in the same wave shape (such as sine wave vs. 1st harmonic only), then the result is exactly the same.

Sine wave is the simplest of all as it contains the fundamental frequency only. The "sharper" the signal shape is, the more harmonics it contains. The biggest source of higher harmonics is a "discontinuity", which you can see in both rectangle and saw waves. In theory, these signals have an infinite number of harmonics. However since our hearing is highly limited to less than 20kHz, the number of harmonics which are relevant is actually pretty small. If you generate a 50Hz signal, which is very low, and assuming that you have extremely good ears and you actually hear 20kHz, then the number of harmonics audible for you is 20000 / 50 = 400.

What happens above 20kHz?

Consider the example above again, what happens with harmonics above 400? These either stay there and simply are not audible, disappear if anti-aliasing is used, or get aliased back under 20kHz in which case you get the typical digital dirt.

When you convert a rectangle wave to harmonics mode, only the first 256 harmonics are used, so it basically works like an infinitely steep low-pass filter. What is the limit then? 50 Hz * 256 = 12.8kHz. The harmonic mode will not produce anything above this limit if you are generating a 50Hz signal. Most people do not hear anything above 15kHz, so this is usually enough, but if not, you may need to use the normal mode where you get the "infinite" number of harmonics.

What you see is not always what you get!

Say you want a rectangle wave and play a 440Hz tone(A4). You would expect the output signal to be a really quick rectangle wave, right? Wrong! If you would do that, and actually most synthesizers on the market do that, you would get the infinite number of harmonics. And, since you are working in say 48kHz sampling rate, the maximum frequency that can actually exist in your signal is 24kHz. So everything above it would get aliased below 24kHz, and there would be a lot of aliased dirt.

The "good" synthesizers perform a so-called anti-aliasing. There are several methods, most of them require quite a lot of CPU or have other limitations. The goal is to remove all frequencies above the 24kHz in our case or in reality, it is more about removing all aliased frequencies above 20kHz - this means, that we do not care about frequencies above 20kHz, because we do not hear them anyway. But we will keep it simple. Let's say we remove everything above 20kHz. You already know that the rectangle wave can be created using an infinite number of harmonics or sine waves. We removed everything above the 45th harmonic (20000 / 440) so our rectangle wave is trying to be formed using just 45 harmonics, so it will not really look like a rectangle wave.

After some additional filtering (like DC removal), the rectangle wave may look completely different than a true rectangle wave, yet it would sound the same! Does it matter? Not really. You simply edit the shape as a rectangle wave and let the synthesizer do the ugly stuff for you. But do not check the output, because it may be very different than what you would expect ;).

How can I generate non-harmonic frequencies?

Ok, so now you are playing a 440Hz (A4) saw wave, it contains 440Hz, 880Hz, 1320Hz etc. Anything generated using the signal generator can contain only these frequencies, the only difference is the levels and phases of each of them. What if you want to make the signal dirty by adding say 500Hz? Well, that is not that simple! Here we are getting into audio synthesizer stuff, so let us just give you a few hints.

The traditional way is to use modulation. One particular method is called frequency modulation (FM). Instead of generating a 440Hz saw wave with your generator, you change the pitch, up and down. You are modulating the frequency, that's why FM. It is basically a vibrato, but as you increase the speed of the vibrato, it gets so quick that you stop noticing the pitch changes (that's very simplified but it serves the purpose) and instead it starts producing a very complex spectrum. Will the 500Hz be there? Well, if setup correctly, yes, but there will also be lots of other non-harmonic frequencies.

Another way is possible without any other tools. Let's say you do not want 440Hz, but 660Hz. Then you may generate 220Hz instead of 440Hz (which is one octave below it) and voila, 660Hz is the 3rd harmonic (3 x 220 is 660)! But you need to shift the saw wave one octave above. Fortunately it is not that hard here - go to the normal mode, select saw tooth, click advanced, and use the harmonics panel to remove the fundamental and leave just the 2nd harmonic, then convert it to harmonic mode. Well, it's not that hard, but it's not exactly simple either...

The only way is, of course, additive synthesis. In that case you do not use one oscillator, but many of them. It lets you generate just about anything. But there is a catch, actually many of them. First, you need to say "ok I want this frequency and that frequency...", the setup is actually infinitely hard as there may be an infinite number of frequencies :). And the second is, of course, CPU requirements.

So is there some ultimate solution? Nope, sorry. The good thing is, you will not probably need it, because while what you see is not always what you get, also what you want is often not what you really want to hear :).

Normal

Normal button switches the generator into the normal mode, which lets you edit the shape of the oscillator. This is especially advantageous for low-frequency oscillators, where the shape matters even though it doesn't have any physical meaning.

Convert

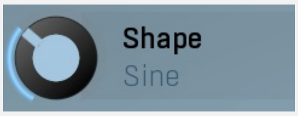
Convert button converts the current shape into harmonic-based representation. Please note that since the number of harmonics is limited, the result will not perfectly resemble the original shape.

Harmonics

Harmonics button switches the generator into the harmonics mode, which lets you edit the levels and phases of individual harmonics. This is especially advantageous for high-frequency oscillators, hence sound generators.

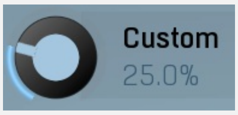
Signal generator in Normal mode

Signal generator in Normal mode works by generating the oscillator shape using a combination of several curves - a predefined set of standard curves, custom shape, step sequencer and custom sample. It also post-processes the shape using several filters including smoothing to custom transformations. This is especially useful when using the oscillator as an LFO (low-frequency-oscillator), where the harmonic contents does not really matter, but the shape does.



Shape

Shape controls the main shape used by the signal generator. There are several predefined shapes: exponential, triangle, sine power 8, sine power 4, sine square, sine, harmonics, more harmonics, disharmonics, sine square root, sine 4 root, rectangle, rect-saw, saw, noise and mess. You can choose any of them or interpolate between any 2 adjacent shapes using this control.



Custom

Custom controls the amount of the custom shape that is blended into the main shape.

Edit

Edit button shows the custom shape editor.

Step

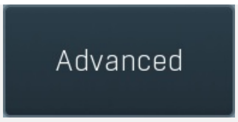
Step controls the amount of the step sequencer shape that is blended into the main shape (which has already been blended with the custom shape).

Edit

Edit button shows the step sequencer editor.

Smooth

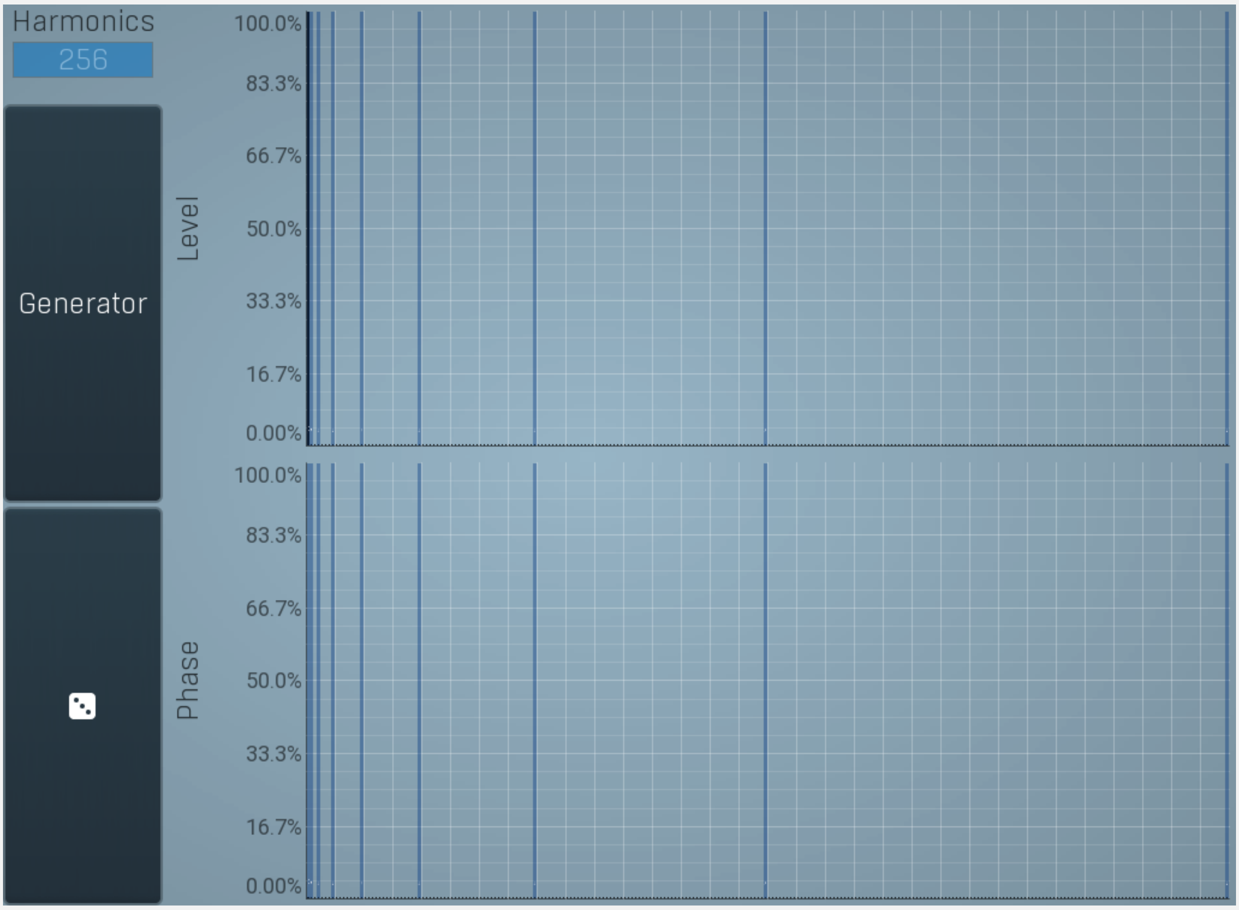
Smooth controls the amount of smoothing. Many shapes, especially those produced by the step sequencer, have rough jagged edges, which may be advantageous, but when used to modulate certain parameters, the output may be clicking or causing other artifacts. Smoothness helps it by smoothing the whole signal shape out and removing these rough edges.



Advanced

Advanced button displays an additional window with more advanced settings for post-processing the signal shape, such as harmonics or custom transformations.

Signal generator in Harmonics mode



Signal generator in Harmonics mode works by generating the oscillator shape using individual harmonics. Essentially a harmonic is a sine wave. The first harmonic, known as the fundamental, fits once in the oscillator time period, hence it is the same as selecting sine wave in the **Normal mode**. The second harmonic fits twice, the third three times etc. In theory, any shape you create in normal mode can be converted into harmonics. However, this approach to signal generation needs an enormous number of harmonics, which is both inefficient to calculate and mostly hard to edit. Therefore, the harmonic mode can process up to 256 harmonics, which is enough for very complex spectrums, however it is still not enough to generate an accurate square wave for example. If your goal is to create basic shapes, it is better to use the normal mode.

It is nearly impossible to say how a particular curve will sound when used as a high-frequency oscillator in a synthesizer, just by looking at its shape. Harmonics mode, on the other hand, is directly related to human hearing and makes this process very simple. In general, the more harmonics you add, the richer the sound will be. The higher the harmonic, the higher the tone. Usually, one leaves the first harmonic enabled too, as this is the fundamental tone, however you may experiment with more dissonant sounds without it.

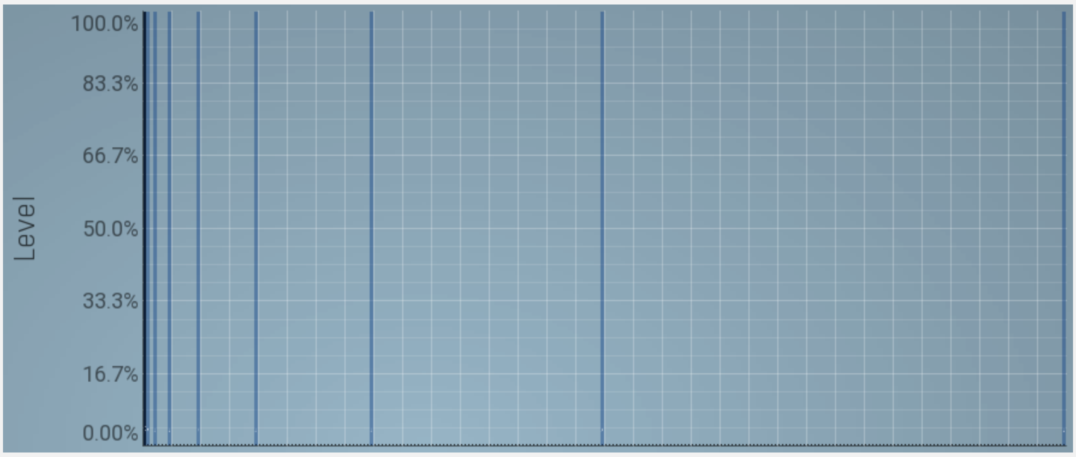
Editing harmonics can be time consuming unless you hear what you want, so a signal generator is also available. This great tool lets you generate a random spectrum by a single click. You can also open the **Generator** settings and edit its parameters, which basically control the audio properties in a more natural way - using parameters such as complexity, harmonicity etc.

Generator

Generator button shows a powerful harmonics generator, which can create unlimited number of various timbres and even analyze a sample and extract harmonics from it.

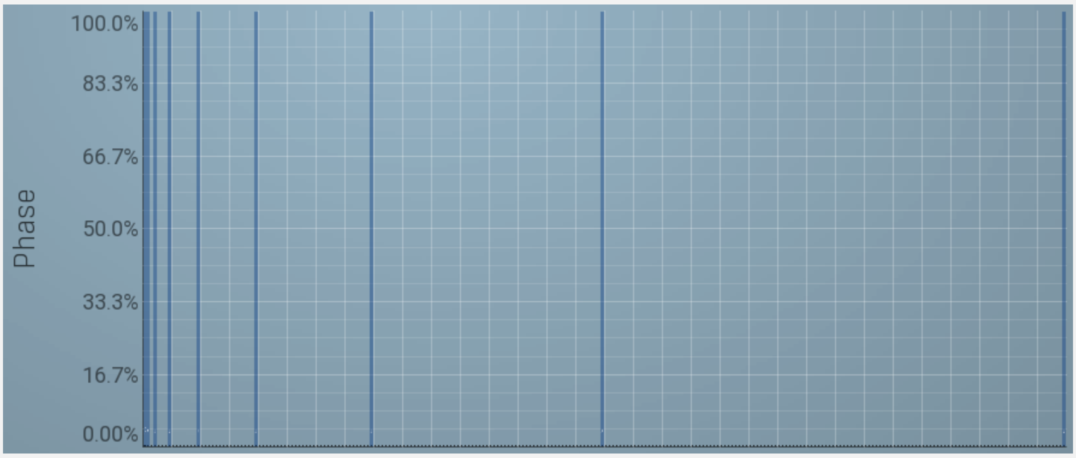
Randomize

Randomize button selects random parameters for the harmonics generator, so you can use it to get a random sound character instantly. Hold **Ctrl** to slightly modify existing generator settings instead of completely changing them.



Magnitudes graph

Magnitudes graph contains the levels of the individual harmonics. The highlighted bars are octaves, thus the 1st, 2nd, 4th, 8th harmonic etc.



Phases graph

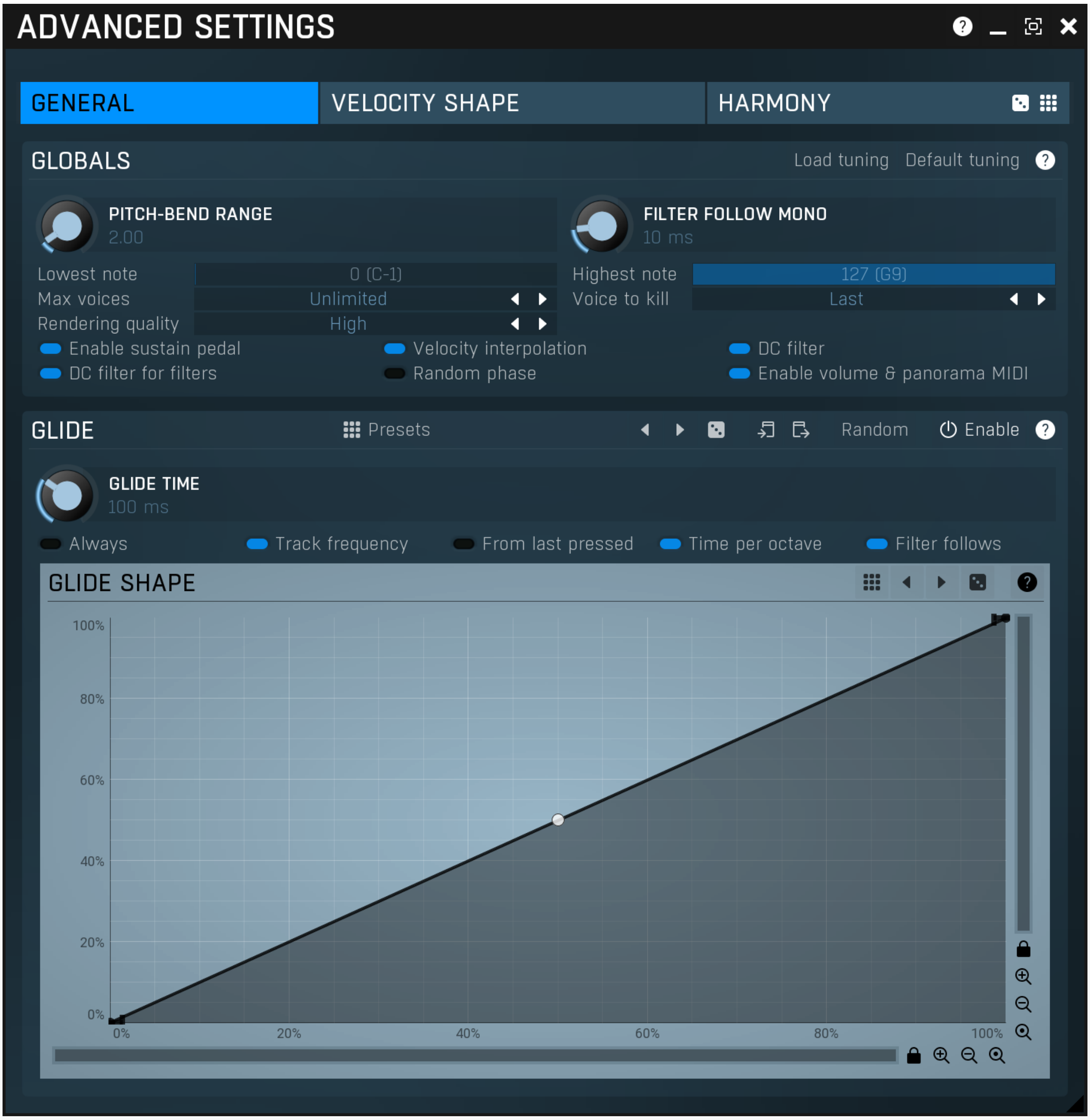
Phases graph contains the phases of the individual harmonics. The highlighted bars are octaves, thus the 1st, 2nd, 4th, 8th harmonic etc.

Show advanced settings

Advanced settings

Advanced settings button displays additional settings.

MPowerSynthAdvanced



Advanced settings window contains more advanced settings, which are used less often and so are intentionally not shown on the main plugin editor.

Tab selector

Tab selector switches between subsections.

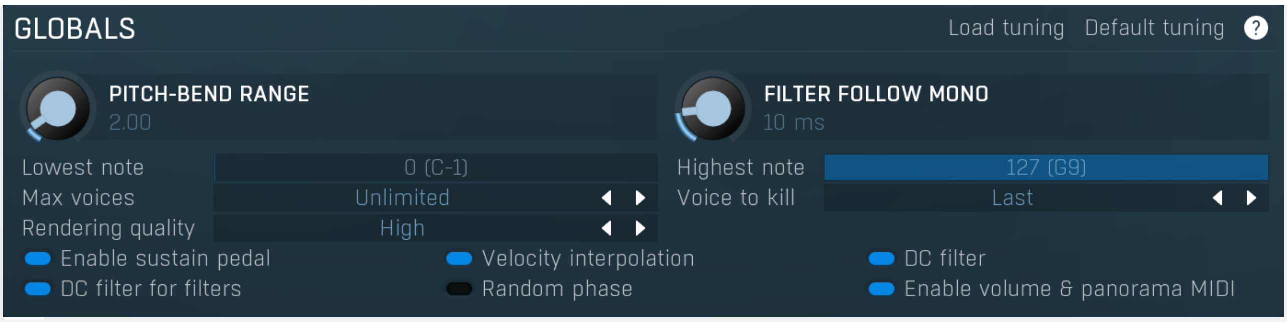
Randomize

Randomize button generates random settings for the tab.

Presets

Presets button chooses a random preset for the tab.

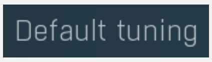
Globals panel



Globals panel contains some global settings controlling the plugin behaviour.

Load tuning

Load tuning button lets you load TUN files containing custom micro-tuning, which will replace the default equal temperament tuning (in which the logarithmic distance between every 2 semitones is exactly the same).

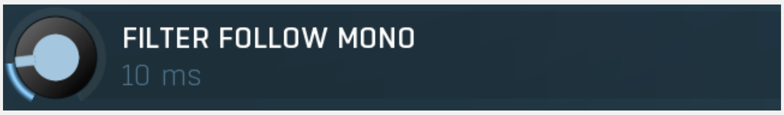


Default tuning

Default tuning button restores the default equal temperament tuning.

Pitch-bend range

Pitch-bend range controls the range in semitones of the global pitch change caused by the MIDI pitch-bend controller.
Range: 0.00 to 24.00, default 2.00



Filter follow mono

Filter follow mono controls the time it takes for the filter to follow note changes in monophonic modes. Use 0ms to make the filter follow instantly, which can cause short glitches. Higher times may provide additional expression, because the filter frequency will highly depend on the first note played.
Range: 0 ms to 10000 ms, default 10.0 ms

Lowest note

Lowest note controls the lowest note that will actually be played. This can be used to make the instrument ignore some notes, so that you can use them to control some features for example.
Range: 0 to 127, default 0

Highest note

Highest note controls the highest note that will actually be played. This can be used to make the instrument ignore some notes, so that you can use them to control some features for example.
Range: 0 to 127, default 127

Max voices

Max voices controls the maximum number of voices that can play at any moment. You can use this to avoid overloading CPU when you play too many voices.

Voice to kill

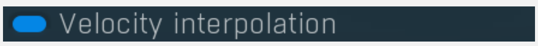
Voice to kill defines which voice is removed, when the number of voices exceeds the limit controlled by **Max voices** parameter.

Rendering

Rendering controls the quality for rendering. If it is better than the **Quality** parameter, it will be used when rendering your project offline (if your host informs the plugin about it). You can use this option if you want to save CPU power, but still want to maximize the audio quality for the final rendering. Be aware that when different settings are used, the output may sound significantly different.

Enable sustain pedal

Enable sustain pedal makes the synthesizer listen to the sustain pedal when processing notes. If this is disabled, the synth ignores the sustain pedal and makes only note-on/off messages relevant.



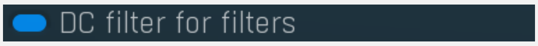
Velocity interpolation

Velocity interpolation activates velocity interpolation for monophonic modes. When this is disabled, only the first note's velocity is relevant and the remaining notes will have the same velocity. If this is enabled however, the velocities of all notes are relevant and a quick transition between them will occur.

DC filter

DC filter activates the global DC filter. It will remove content below 20Hz, which isn't audible but is interfering with many processing algorithms. Please note, that when the DC filter is use, the rendered oscillator shapes may start looking significantly different -

square waves won't look like square waves at all. This is not an error, the generators will still sound the same, just the content below 20Hz will be removed, which is what causes the signal shape difference.



DC filter for filters

DC filter for filters activates the DC filter for output of each filter. It will remove content below 20Hz, which isn't audible but is interfering with many processing algorithms. Such content is often created by the distortion algorithms used by the filters.

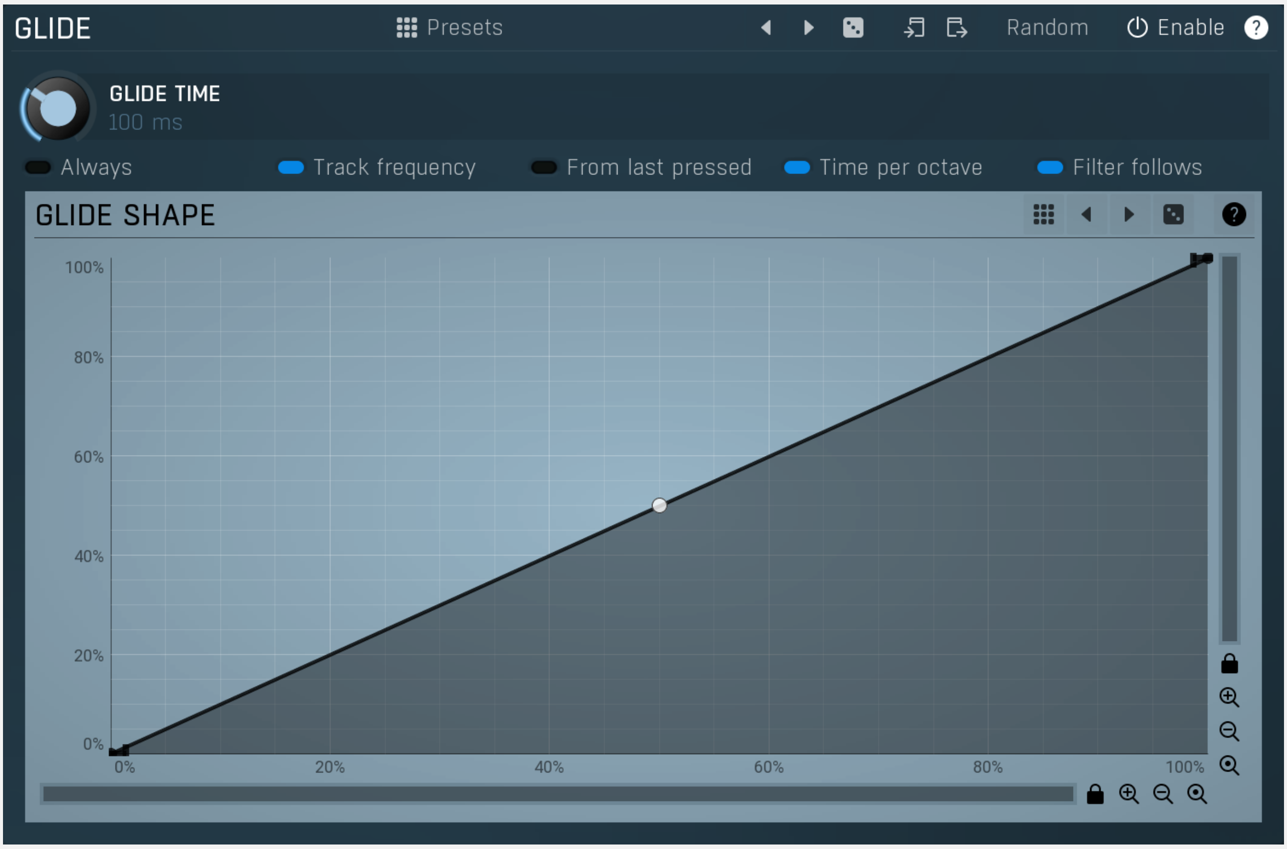
Random phase

Random phase activates the random initial phase mode for all oscillators, which kind of simulates the behaviour of analog synthesizers, where each note is just slightly different. Please note that if **Unison** is enabled, this option will be ignored and the unison's settings will be used.

Enable volume & panorama MIDI

Enable volume & panorama MIDI makes the plugin automatically follow the volume and panorama MIDI messages. You may want to disable it if your MIDI controller malfunctions for example.

Glide panel



Glide panel contains parameters of the glide feature, which lets new notes glide from previous ones.

Presets

Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.

Left arrow

Left arrow button loads the previous preset.

Right arrow

Right arrow button loads the next preset.

Randomize

Randomize button loads a random preset.

Copy

Copy button copies the settings onto the system clipboard.

Paste
Paste button loads the settings from the system clipboard.

Random
Random button generates random settings using the existing presets.

Glide time
Glide time controls the length of the glide.
Range: 1.0 ms to 10000 ms, default 100 ms

Always
Always starts the glide every time you press a note. If this is disabled, the glide is performed only if you currently hold another note, hence the glide is started only if the 2 notes are overlapping.
For example, if you hold key C and press key G, a glide is always performed. But if you press C, release it and then press G, the glide is started only if this parameter is enabled.

Track frequency
Track frequency overrides the behaviour when the notes are followed by the glide and instead it keeps track of the current frequency, so it 'starts where the last one ended'. This is useful for long glides. It even works with polyphonic glides, where at any moment the sound can split from the same pitch etc.
For example, if the glide length is 10 seconds and you press C0 and C4 afterwards for a short time, the glide doesn't have enough time to get to C4. Normally next time the glide would start directly at C4 even though it didn't manage to get to that pitch. With this option enabled, the glide will continue where it finished (a frequency somewhere between C0 and C4).

From last pressed
From last pressed makes the glide start from the last key pressed, which may actually not be the one you are holding right now.
For example, if you hold key C, then press and release key E (so there's a short glide from C towards E) and then press key G, there are 2 options. First, the glide may go from C to G as C is currently pressed, and that's what happens if 'from last pressed' is disabled. Second, the glide may go from E to G, because you most recently pressed E, and that's what happens with this option enabled.

Time per octave
Time per octave makes the glide time depend on the pitch distance (interval) between the 2 notes. When enabled, the glide time specifies the time needed to glide across one octave. If not enabled, the glide time is the total time to get from the initial note to the target. In that case the actual glide speed differs depending on the 2 notes, because it takes the same amount of time to glide between any 2 notes, no matter if they are 1 semitone or 4 octaves away from each other.

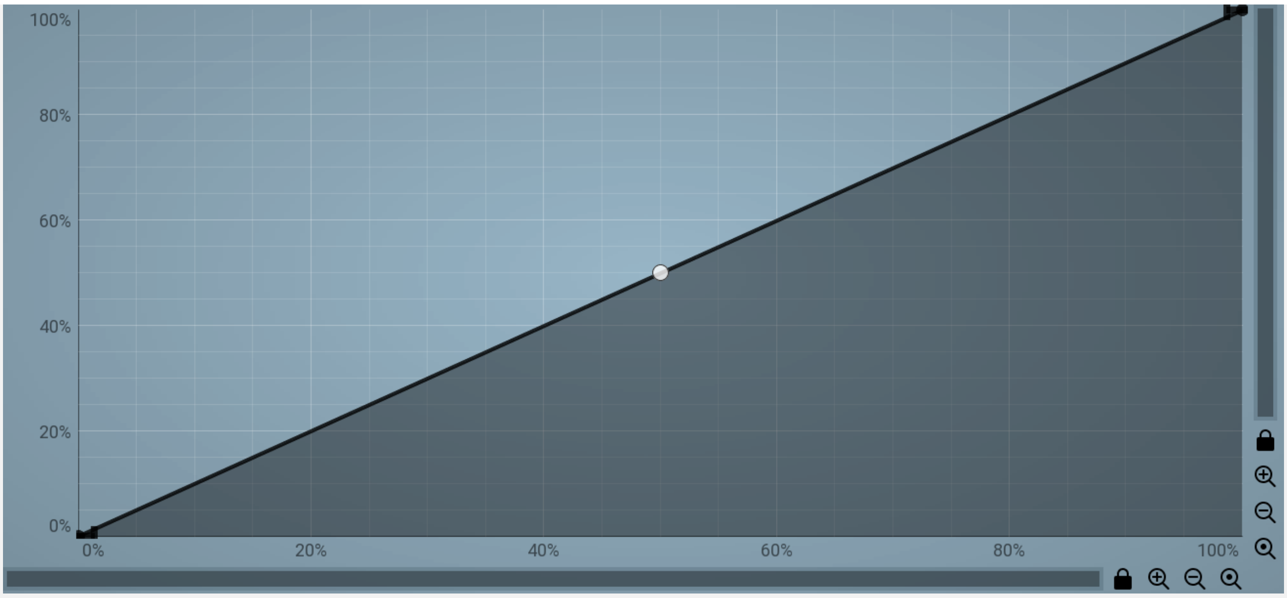
FromFilter follows
FromFilter follows makes the filters follow the glide pitch. If this is disabled, the filters use the frequencies of the target notes and ignore the glide, which can be useful for example when you want the note to glide towards the target note, which will resonate using the filter.

Presets
Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.

Left arrow
Left arrow button loads the previous preset.

Right arrow
Right arrow button loads the next preset.

Randomize
Randomize button loads a random preset.



EnvelopeEditorGraph

Envelope graph

Envelope graph provides an extremely advanced way to edit any kind of shape that you can imagine. An envelope has a potentially unlimited number of points, connected by several types of curves with adjustable curvature (drag the dot in the middle of each arc) and the surroundings of each point can also be automatically smoothed using the smoothness (horizontal pull rod) control. You can also literally draw the shape in drawing mode (available via the main context menu).

- **Left mouse button** can be used to select points. If there is a *point*, you can move it (or the entire selection) by dragging it. If there is a *curvature circle*, you can set up its tension by dragging it. If there is a *line*, you can drag both edge points of it. If there is a *smoothing controller*, you can drag its size. Hold **Shift** to drag more precisely. Hold **Ctrl** to create a new point and to remove any points above or below.
- **Left mouse button double click** can be used to create a new point. If there is a *point*, it will be removed instead. If there is a *curvature circle*, zero tension will be set. If there is a *smoothing controller*, zero size will be set.
- **Right mouse button** shows a context menu relevant to the object under the cursor or to the entire selection. Hold **Ctrl** to create or remove any points above or below.
- **Middle mouse button** drag creates a new point and removes any points above or below. It is the same as holding Ctrl and dragging using left mouse button.
- **Mouse wheel** over a point modifies its smoothing controller. If no point is selected, then all points are modified.
- **Ctrl+A** selects all points. **Delete** deletes all selected points.

Presets

Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.

Left arrow

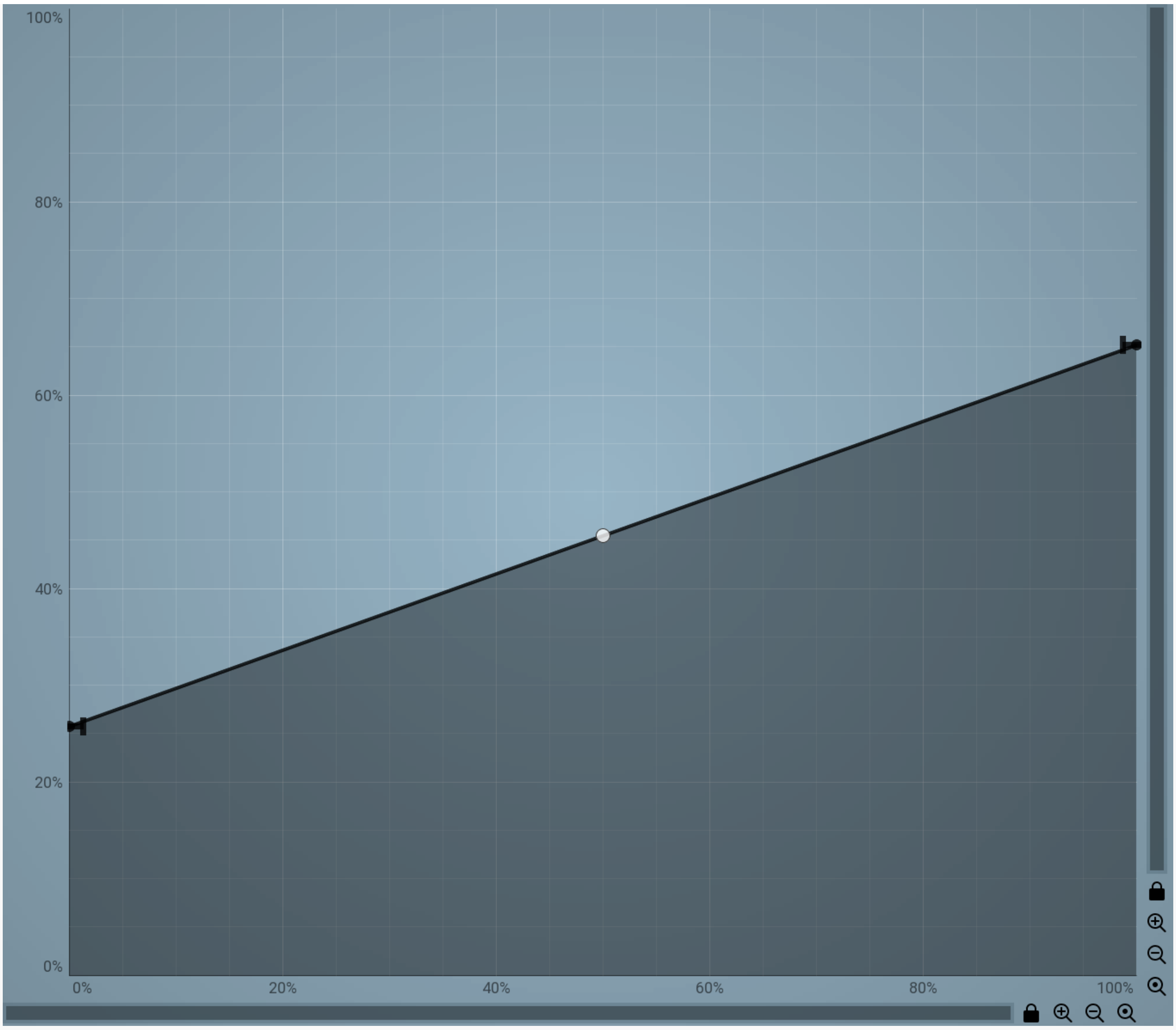
Left arrow button loads the previous preset.

Right arrow

Right arrow button loads the next preset.

Randomize

Randomize button loads a random preset.



EnvelopeEditorGraph

Envelope graph

Envelope graph provides an extremely advanced way to edit any kind of shape that you can imagine. An envelope has a potentially unlimited number of points, connected by several types of curves with adjustable curvature (drag the dot in the middle of each arc) and the surroundings of each point can also be automatically smoothed using the smoothness (horizontal pull rod) control. You can also literally draw the shape in drawing mode (available via the main context menu).

- **Left mouse button** can be used to select points. If there is a *point*, you can move it (or the entire selection) by dragging it. If there is a *curvature circle*, you can set up its tension by dragging it. If there is a *line*, you can drag both edge points of it. If there is a *smoothing controller*, you can drag its size. Hold **Shift** to drag more precisely. Hold **Ctrl** to create a new point and to remove any points above or below.
- **Left mouse button double click** can be used to create a new point. If there is a *point*, it will be removed instead. If there is a *curvature circle*, zero tension will be set. If there is a *smoothing controller*, zero size will be set.
- **Right mouse button** shows a context menu relevant to the object under the cursor or to the entire selection. Hold **Ctrl** to create or remove any points above or below.
- **Middle mouse button** drag creates a new point and removes any points above or below. It is the same as holding Ctrl and dragging using left mouse button.
- **Mouse wheel** over a point modifies its smoothing controller. If no point is selected, then all points are modified.
- **Ctrl+A** selects all points. **Delete** deletes all selected points.

Harmony panel

Harmony panel controls additional voices created virtually by the synth. When you press a note, the plugin may behave as if you had pressed additional notes with different velocities, tuning, delays etc.

Presets

Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.

Left arrow

Left arrow button loads the previous preset.

Right arrow

Right arrow button loads the next preset.

Randomize

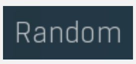
Randomize button loads a random preset.

Copy

Copy button copies the settings onto the system clipboard.

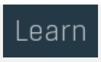
Paste

Paste button loads the settings from the system clipboard.



Random

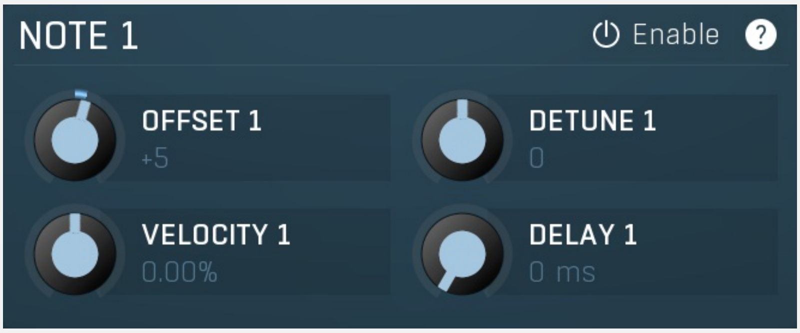
Random button generates random settings using the existing presets.



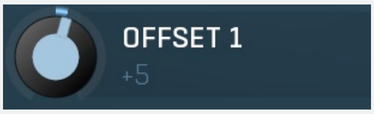
Learn

Learn button lets you learn the whole harmony by playing it. Enable the button, then press a chord on your keyboard and the plugin will configure the harmony settings so that the plugin will play the chord when one key is pressed.

Note panel

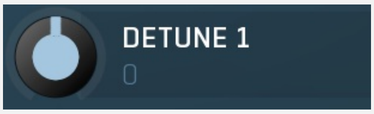


Note panel contains parameters of a virtual note. If you enable the note and press a key, the synth will behave as if you had pressed another key at the same time, with different velocity, tuning, delay etc. Note that the keys cannot be duplicates, so if you for example hold C1 and there is a virtual note 12 semitones up (C2) both are played, but when you then press C2 as well, this one won't generate a new sound and will be ignored as it is already playing, but C3 will be added as well.



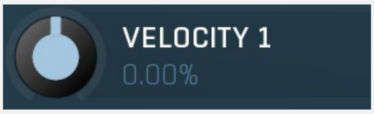
Offset

Offset controls the offset of the virtual note from the original one in semitones.
Range: -48 to +48, default +5



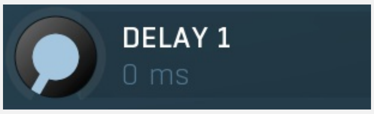
Detune

Detune controls the detuning of the virtual note in cents. The actual pitch difference is the sum of these two control values.
Range: -100.00 to +100.00, default 0



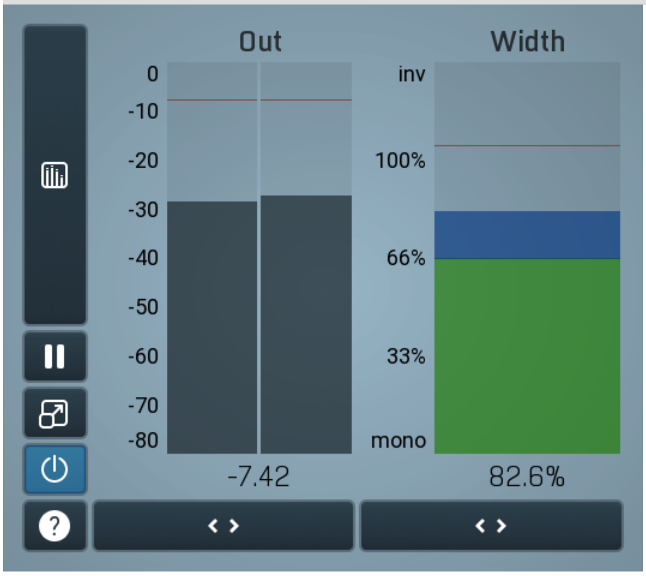
Velocity

Velocity controls the difference in velocity of the virtual note from the original one.
Range: -100.0% to 100.0%, default 0.00%



Delay

Delay makes the virtual note delayed from the original one and can be used for strumming effects for example.
Range: 0 ms to 1000 ms, default 0 ms



Global meter view

Global meter view provides a powerful metering system. If you do not see it in the plug-in, click the **Meters** or **Meters & Utilities** button to the right of the main controls. The display can work as either a classical level indicator or, in time graph mode, show one or more values in

time. Use the first button to the left of the display to switch between the 2 modes and to control additional settings, including pause, disable and pop up the display into a floating window. The meter always shows the actual channels being processed, thus in M/S mode, it shows mid and side channels.
In the classical level indicators mode each of the meters also shows the recent maximum value. Click on any one of these values boxes to reset them all.

Out meter indicates the total output level. The output meter is the last item in the processing chain (except potential downsampling and other post-processing). It is always recommended to keep the output under 0dB.

As the levels approach 0dB, that part of the meters is displayed with **red** bars. And recent peak levels are indicated by single bars.

Width meter shows the stereo width at the output stage. This meter requires at least 2 channels and therefore does not work in mono mode. Stereo width meter basically shows the difference between the mid and side channels.
When the value is **0%**, the output is monophonic. From 0% to 66% there is a green range, where most audio materials should remain.
From 66% to 100% the audio is very stereophonic and the phase coherence may start causing problems. This range is colored blue. You may still want to use this range for wide materials, such as background pads. It is pretty common for mastered tracks to lie on the edge of green and blue zones.
Above 100% the side signal exceeds the mid signal, therefore it is too monophonic or the signal is out of phase. This is marked using red color. In this case you should consider rotating the phase of the left or right channels or lowering the side signal, otherwise the audio will be highly mono-incompatible and can cause fatigue even when played back in stereo.
For most audio sources the width is fluctuating quickly, so the meter shows a 400ms average. It also shows the temporary maximum above it as a single coloured bar.
If you right click on the meter, you can enable/disable loudness pre-filtering, which uses EBU standard filters to simulate human perception. This may be useful to get a more realistic idea about stereo width. However, since humans perceive the bass spectrum as lower than the treble, this may hide phase problems in that bass spectrum.

Time graph
Time graph button switches between the metering view and the time-graphs. The metering view provides an immediate view of the current values including a text representation. The time-graphs provide the same information over a period of time. Since different time-graphs often need different units, only the most important units are provided.

Pause
Pause button pauses the processing.

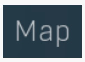
Popup
Popup button shows a pop-up window and moves the whole metering / time-graph system into it. This is especially useful in cases where you cannot enlarge the meters within the main window or such a task is too complicated. The pop-up window can be arbitrarily resized. In metering mode it is useful for easier reading from a distance for example. In time-graph mode it is useful for getting higher accuracy and a longer time perspective.

Enable
Enable button enables or disables the metering system. You can disable it to save system resources.

Collapse
Collapse button minimizes or enlarges the panel to release space for other editors.

Collapse
Collapse button minimizes or enlarges the panel to release space for other editors.

Utilities



Map

Map button displays all current mappings of modulators, multiparameters and MIDI (whichever subsystems the plugin provides).

Multiparameter

Multiparameter button displays settings of the multiparameter. The multiparameter value can be adjusted by dragging it or by pressing Shift and clicking it to enter a new value from the virtual keyboard or from your computer keyboard.

Click on the button using your left mouse button to open the **Multiparameter** window where all the details of the multiparameter can be set. Click on it using your right mouse button or click on the **menu button** to the right to display an additional menu with learning capabilities - as described below.

Menu

Menu button shows the **smart learn** menu. You can also use the right mouse button anywhere on the multiparameter button.

Learn attaches any parameters, including ranges. Click this, then move any parameters through the ranges that you want and click the multiparameter button again to finish. While learning is active, "REC" is displayed on the multiparameter button and learning mode is ended by clicking the button again.

Clear & Learn clears any parameters currently in the list then attaches any parameters, including ranges. Click this, then move any parameters through the ranges that you want and click the multiparameter button again to finish. While learning is active, "REC" is displayed on the multiparameter button and learning mode is ended by clicking the button again.

Reset resets all multiparameter settings to defaults.

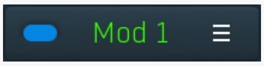
Quick Learn clears any parameters currently in the list, attaches one parameter, including its range and assigns its name to the multiparameter. Click this, then move one parameter through the range that you want.

Attach MIDI Controller opens the MIDI Settings window, selects a unused parameter and activates MIDI learn. Click this then move the MIDI controller that you want to assign.

Reorder to ... lets you change the order of the multiparameters. This can be useful when creating active-presets. Please note that this feature can cause problems when one multiparameter controls other multiparameters, as these associations will not be preserved and they will need to be rebuilt.

In learning mode the multiparameter does not operate but rather records your actions. You can still adjust every automatable parameter and use it normally. When you change a parameter, the plugin associates that parameter with the multiparameter and also records the range of values that you set.

For example, to associate a frequency slider and make a multiparameter control it from 100Hz to 1KHz, just enable the smart learn mode, click the slider then move it from 100Hz to 1KHz (you can also edit the range later in the Multiparameter window too). Then disable the learning mode by clicking on the button.



Modulator
Modulator button displays settings of the modulator. It also contains a checkbox, to the left, which you can use to enable or disable the modulator. Click on it using your right mouse button or use the **menu button** to display an additional menu with learning capabilities - as described below.

Menu
Menu button shows the **smart learn** menu. You can also use the right mouse button anywhere on the modulator button.

Learn activates the learning mode and displays "REC" on the button as a reminder, **Clear & Learn** deletes all parameters currently associated with the modulator, then activates the learning mode as above. After that every parameter you touch will be associated to the modulator along with the range that the parameter was changed. Learning mode is ended by clicking the button again.

In smart learn mode the modulator does not operate but rather records your actions. You can still adjust every automatable parameter and use it normally. When you change a parameter, the plugin associates that parameter with the modulator and also records the range of values that you set.

For example, to associate a frequency slider and make a modulator control it from 100Hz to 1KHz, just enable the smart learn mode, click the slider then move it from 100Hz to 1KHz (you can also edit the range later in the modulator window too). Then disable the learning mode by clicking on the button.

Menu
Menu button displays additional menu containing features for modulator presets and randomization.

Randomize
Randomize button generates random arpeggiator settings.

Lock
Lock button displays the settings of the global parameter lock. Click on it using your left mouse button to open the Global Parameter Lock window, listing all those parameters that are currently able to be locked.
Click on it using your right mouse button or use the **menu button** to display the menu with learning capabilities - **Learn** activates the learning mode, **Clear & Learn** deletes all currently-lockable parameters and then activates the learning mode. After that, every parameter you touch will be added to the lock. Learning mode is ended by clicking the button again.
The On/Off button built into the Lock button enables or disables the active locks.

Collapse
Collapse button minimizes or enlarges the panel to release space for other editors.