

# MSoundFactory



## Overview

MSoundFactory is a modular instrument featuring 2 separate sections, the **Generator** and the **FX**, together with a general **Globals** tab.

## Generator and Globals

The **Generator section** is a modular voice generator. In this context a "voice" is the audio produced when a MIDI note is received; a 3-note chord will result in three separate voices. You can use various synthesizers, samplers, filters and even per-voice effects here and this part is processing each voice separately. The processing of each voice can result in 1 to 6 output streams, depending of the routing between the processing modules. Each of these 6 outputs (for all current voices) is routed separately to the FX section. When all voices are rendered (that is output from MSoundFactory), they are all mixed together. You can use the single output edition of MSoundFactory, or the 6-output one (which would enable you to apply different external effects to the different parts of the voices).

Before we explain the FX section, we also need to discuss the main MSoundFactory tab, the **Globals**. While global pitch, volume, panorama and quality are pretty self-explanatory, there are 2 vital parts - voicing mode and global envelope - to discuss. **Voicing mode** controls whether the instrument is polyphonic or monophonic, or operates as a (one-shot) trigger which is useful for percussive sounds. The **global envelope** has 2 effects - firstly it is applied on each of the 6 Generator outputs of each voice before they are mixed and routed to that input in the FX section. And secondly, it defines when a voice ends - once the envelope ends, the voice is finished. The envelope is a very flexible DAHDSR envelope, with several parameters that can be modulated per-voice. Also note that, in the trigger voicing mode, the Sustain section is ignored, with the envelope going from the Decay stage to the Release stage. This is important to understand in relation to the per-voice effects, see below.

## FX

The **FX section** has 6 inputs, matching the 6 outputs of the Generator. The FX section is therefore always processing all voices mixed together, but the 6 audio streams let you process parts of the voice output differently. For example, you may place a drum sampler module in the Generator's lane 1 for initial sound attack and a wavetable module in lane 2. The voice is the audio from both lanes. Then placing a reverb on lane 1 in the FX section would be processing only the drum sampler output.

## Per-voice effects

There are effects in the Generator section as well, these are the per-voice FX. Such a feature is rarely available in instrument plug-ins and it is important to understand what is this for and what the differences are between global FX and per-voice FX.

First of all, per-voice effects process each voice individually. As a consequence the CPU consumption is higher (unless only one voice is played) compared with using the same FX in the FX section. For our 3-note chord, there would effectively be 3 instances of the per-voice effect compared with one in the FX section.

Some of their parameters are available for per-voice modulation (see below). Per-voice FX are inserted on a lane (one of the six audio streams), they can be applied in parallel to the generated audio and produce quite different outputs. And finally, their results may be quite different compared with using the global effects, whenever multiple notes are played at the same time.

Some effects won't make a difference - all equalizers, static filters and non-modulated reverbs for example. These effects are not creating any new frequency content and to spare you complex mathematical details, let's just say that it doesn't matter in which order you use them. In other words EQ->reverb will produce the same output as reverb->EQ. The same is true for mixing, it doesn't matter if you mix track 1 with track 2 and then mix the result with track 3 or do it in a different order. The conclusion here is that if you only want to use a static filter without any modulation at the end of the Generator pipeline for example, then it is better to put it into the FX section, because the result will be the same, but it would require less CPU. It's not true for a reverb though, because the global envelope is applied AFTER the generator. So placing a reverb into the Generator section could result in the reverb tail being cut off by the global envelope.

Most effects are not like that; they produce a new spectral content and it does matter in which order you process them. The most obvious one is distortion, of any kind. Processing a sine wave using a distortion doesn't result in a sine wave anymore, but in a more complex spectrum. But there's a problem - processing 2 sine waves (s1 and s2) mixed together using a distortion doesn't produce the same thing as processing 2 sine waves through a distortion then mixing the results.

*In fact, distorting a mixed signal can quickly get out of hand. In our case with 2 mixed sine waves processing it with a distortion would indeed produce the distorted s1 and distorted s2, but also some complex output featuring additional frequencies such as s1+s2 and s1-s2. With signals more complex than just sine waves the results can be quite brutal. This is why playing complex chords on highly distorted electric guitars is rarely possible, since the output would quickly become dissonant. And that's why the so-called powerchords (a fifth) are exploited so much on electric guitar - the fifth nicely aligns most harmonics, so the complex output is still not dissonant, for the most part.*

A lesson here is that using a distortion on the FX tab may sound good, but it is likely to become ugly and dissonant once you play more than a single note. But distortion is an extremely handy tool for sound design, so the per-voice FX give you a solution - these let you distort each note separately, avoiding all this madness. Unless, of course, that is what you're aiming for.

Of course the per-voice FX let you do quite a lot more than that. For instance, you can modulate the amount of flanging by an envelope. Or you can process an oscillator by one effect and then merge the results using frequency modulation with another oscillator. And... well there are pretty much an infinite number of possibilities.

## Dual modulation system

MSoundFactory features a dual modulation system, a **global** and a **per-voice** one. Each of them is useful in different situations, but for most sound design situations you are likely to exploit the per-voice modulation system.

**Global modulation** system is available via the classic modulators that you know from other MeldaProduction plugins. These can be applied to nearly any parameter in the Globals, Generator or FX sections, and the changes affect all voices identically. Each of the modulators can modulate several parameters.

**Per-voice modulation** is available only for some of the parameters in the Generator section and in the global envelope, but these are processed individually for each voice. For example, you can use these to evolve a filter frequency over time. At the bottom of the **Generator** tab you can see the Per-voice modulation sources panel, where you can configure some of these sources; the other per voice modulation sources are not available here, because there is nothing to configure for them (for example, MIDI controllers, MIDI note properties etc).

Each parameter in the generator plug-in modules that can be modulated has a button just to the left of it. It displays the modulation range and the current value (which works well only if there is just a single voice playing of course). You can simply drag & drop the arrow pointer on any modulation source onto that modulation button. But there's more. Clicking the left-hand half of the modulation button opens a modulation settings window, which contains a 4x4 modulation grid. So every parameter can be controlled by up to 16 modulation sources. Sources on each row in the modulation grid are combined using selectable operators and the rows are summed together to form the resulting modulation value. The depth of each modulation source and the overall modulation Depth can also be set.

That sounds a bit techy, but it's actually pretty simple, let's see it in action in an example: Click on the modulation button of any parameter you'd like to modulate. In the modulation settings only one item of the grid is available, the top left one (also known as "A1"), it is labelled "Off". Click on it and select LFO 1. Increase the overall modulation depth, nothing happens! Now, play a note on your MIDI keyboard so that it starts doing something. You'll hear a sound, see the horizontal bars move in the modulation settings window and see the modulation amount change on the modulation button.

Now the parameter is modulated by the LFO 1. But also 2 new grid items are now available, the second one on the first row ("A2") and the first one of the second row ("B1"). Click on the first item of the 2nd row ("B1") and select Env 1. Now the parameter is modulated by LFO 1 and Env 1 and the outputs of each are summed, which means they do not affect each other. From the beginning of every voice the LFO is modulating the parameter fully, same for Env 1.

Now click on the Env 1 field again and select "Off" to remove the modulation source. Then select Env 1 for the 2nd item on the 1st row instead. Now the parameter is once again controlled by both modulation sources, but not in quite the same way - to see it, well, increase the attack of the envelope Env 1. Press a key and see how initially the LFO is not doing much and in time its effect is strengthening. That's the case when the sources are multiplied. The first source on a row defines the modulation, and the remaining sources on that row say "how much". You can change the multiplication of the two sources to addition, subtraction, maximum or minimum too. Besides the obvious evolution over time that you can get from this, you can also control the amount of modulation by key pressure, and other controllers for example. The available sources are all listed in the modulation source selection window.

Finally, the horizontal blue bars below each modulation source in the grid show the value of that source, red bars below the row show that

combined value of the active sources in that row and the green bar just above the 4x4 grid shows the overall value (and that is the one that actually modulates the parameter).

## Per-voice modulation sources

In this section we are going to explore the various modulation sources available in the Generator. Click any modulation button to see the pre-voice modulation matrix, then click any available modulator button in the 4x4 grid to see the Select modulation source window.

## Controllers section

Controllers are essential for expressive playing, now even more with "3D" (MIDI Polyphonic Expression, or "MPE") instruments such as Linnstrument or Seaboard. By default only the note velocity and pitch-bend are used by the instrument, so every other MIDI controller needs to be mapped in some way if you want to use it. In MSoundFactory we decided to make some of the controllers mandatory for all devices, simply to make everything MPE compliant, and so that everyone with any MPE controller can take advantage of the modern expressive playing.

Note that although the following list doesn't include all the controllers, it does include all the standard per-voice controllers. You can also use global MIDI settings in MSoundFactory to map other controllers to any parameters as well. It wouldn't be "per-voice", nor would most of these controllers. Having said that, **Main controllers** can be defined in MIDI settings and marked as "MPE".

### Aftertouch & Pressure (mandatory)

While velocity defines how quickly the user initially presses a key, pressure is sent by a MIDI controller afterwards and depends on how strongly the user presses the pressure pads under each key. This enables somewhat expressive playing when the user varies the pressure under each key.

### Breath CC#2

Breath controllers are not that popular these days, but the principle is self-explanatory - you control it by breathing into a tube attached to your instrument. It's often used by MIDI controllers simulating wind instruments such as a saxophone or flute, where you literally breathe into them to make them sound, just like with the real deal.

### Expression CC#11

Expression is usually available as a dedicated pedal attached to the MIDI controller. Imagine a guitarist's wah-wah pedal for example.

### Modulation wheel CC#1 (mandatory)

Modulation wheel is available in nearly all MIDI keyboards and hence it is often considered mandatory for expressive playing. Some MPE instruments can also send this as one of the 3D clues.

### Pitch-bend

Pitch-bend usually modifies pitch and in MPE controllers it is actually used for gliding between notes. It is usually not a good idea to remove the default pitch-bending effect by setting the **Pitch-bend range** to 0, but you can use this controller to make the instrument do more than just pitch change.

### Timbre CC#74 (mandatory)

Timbre is used by MPE controllers to send information about vertical movement on the player's finger on the key.

## Envelope section

Envelope section contains several custom envelopes and the global envelope. A general rule of thumb - use as few of them as possible. These envelopes are "calculated" only if they are attached to some parameters (except for the global envelope, which needs to be calculated every time of course), so in order to save CPU it is better to avoid using several identical envelopes for example.

## LFO section

LFO section contains several LFOs (low-frequency oscillators) and Random sources. Both kinds have various speed settings and they mainly

differ in the fact that LFOs are oscillators, and are repeating the same shape all the time, while Random sources are generating a pseudo-random flow, so they are useful for mimicking analog behaviour for example.

**LFOs** have quite a huge range of up to 20 kHz, so these are technically not LFOs anymore, but general oscillators. You can therefore use them as classic LFOs, or they can follow the key frequency for example. That can be useful for some creative sound design applications. However be careful, sound-quality-wise it is always better to use a dedicated module for specific tasks. For example, you can use an Oscillator module, or you can use a Set module and modulate its output value by the LFO. In this case it is obvious, that the Oscillator module will produce a much better audio quality as it is designed for this specific purpose and will completely avoid aliasing, while the Set module wouldn't. Global oversampling could help though.

## Main Controllers section

Main controllers section is similar to the **Controllers section**. However you can consider the 16 Main controllers to be virtual controllers, they are mapped to real MIDI CC controller numbers in MIDI settings. These are sort of proxies to the actual MIDI controllers, so that users can set the Main controllers depending on the MIDI controller they are using and changing map them to the target parameters using the Main controllers. Then, when a user changes the MIDI controller for example, all he needs to do is change the Main controllers.

Main controllers can also be MPE. This means that MSoundFactory will recognize different values on each MIDI channel for that controller. By selecting such a Main controller in the per-voice modulation, each voice can be controlled differently. This option should be used **ONLY** for custom MPE controllers which have the per-voice controllers available.

## Note section

Note section contains many parameters of the note being played or even some specific sources, which are different for each note.

### Note frequency

Note frequency follows the frequency of the voice, including glide. Since modulation always requires a range of 0 to 1, this controller follows the path of highest compatibility by transforming the logarithmic range of 20 Hz to 20 kHz into the 0 to 1 range. Since most frequency parameters follow the same range, this should make everything nicely compatible with each other.

### Note key

Note key is another parameter following pretty much the same thing. It follows the key number (0 to 127 by MIDI standard) and transforms it into the 0 to 1 modulation range.

### Note random 1 & 2

Both produce a random value for each note. They are good for making each voice slightly different, just like the analog devices would. If you set say Note random 1 to modulate multiple parameters, they will all be modulated by exactly the same amount each time.

### Note sequence 1 & 2

Both produce a random sequence, which is the same for every note received. This is useful when using the Unison module for example, which creates several voices with varying pitch, but its main purpose is to make each voice different by more than just the pitch. That's where these 2 sources come in handy. By assigning them to modulate various parameters you will make those parameters different for each unison voice, but overall the main voice will be the same every time.

### Note true random

This one is similar to **Note random**, however it generates a different value for every parameter to which you assign it.

### Note velocity

Velocity may do more than just control volume. It is defined by how quickly the user presses the key.

### Note velocity off

This one is initially 0 and as soon as user releases the key, it becomes defined by how quickly he/she does that. If the controller supports it of course.

## Special section



Here you can find some more modulation sources, which don't really have a specific category.

## Gate

Gate is 1 initially and becomes 0 as soon as the user releases the key.

## Noise 1 & 2

Both produce a noise signal, which can be used creatively, usually to produce some sort of distortion. If you set say Noise 1 to modulate multiple parameters, they will be modulated in exactly the same way.

## Random on note

This is not strictly a per-voice modulation source, since it has the same value for all voices. It holds a random value, which gets changed every time the user presses a key. Be aware that this makes the amount of modulation identical for all voices.

## True noise

Produces a noise signal, which (unlike **Noise 1 & 2**) is different for every parameter to which you assign it.

## Advanced settings

If you feel that a classic feature on the main plugin window is missing, you will probably find it in the Advanced settings available by clicking the **Show advanced settings** button just above the global envelope in the Globals section. There you can find settings controlling how the instrument assigns voices, whether it follows sustain and other controllers, you can configure the glide (portamento) feature, transform the velocity and other controllers, and finally exploit the harmony feature, which lets you play chords by pressing a single key.

## Easy screen vs. Edit screen

The plugin provides 2 user interfaces - an **easy screen** and an **edit screen**. Use the Edit button to switch between the two.

By default most plugins open on the **easy screen** (edit button released). This screen is a simplified view of the plugin which provides just a few controls. On the left hand side of the plugin you can see the list of available **devices / instruments** (previously called 'active presets'), that is, presets with controls. These controls are actually nothing more than multiparameters (single knobs that can control one or more of the plug-in's parameters and sometimes known as Macro controls in other plug-ins) and are described in more detail later. Each device may provide different controls and usually is intended for a specific purpose. The easy screen is designed for you to be able to perform common tasks, quickly and easily, without the need to use the advanced settings (that is, those available on the Edit screen).

In most cases the devices are highlighted using different text colors. In some cases the colors only mark different types of processing, but in most cases the general rule is that **black/white devices** are the essential ones designed for general use. **Green devices** are designed for a specific task or audio materials, e.g. de-essing or processing vocals in a compressor plugin. **Red devices** usually provide some very special processing or some extreme or creative settings. In a distortion plugin, for example, these may produce an extremely distorted output. **Blue devices** require an additional input, a side-chain or MIDI input usually. Without these additional inputs these **Blue** presets usually do not function as intended. Please check your host's documentation about routing side-chain and MIDI into an effect plugin.

To the right of the controls are the meters or time-graphs for the plugin; the standard plugin Toolbar may be to the right of these or at the bottom of the plugin.

By clicking the **Edit button** you can switch the plugin to **edit mode** (edit button pushed). This mode provides all the of the features that the plugin offers. You lose no settings by toggling between edit mode and the easy screen unless you actually change something. This way you can easily check what is "under the hood" for each device, or start with an device and then tweak the plugin settings further.

Devices are factory specified and cannot be modified directly by users, however you can still make your own and store them as normal presets. To do so, configure the plugin as desired, then define each multiparameter and specify its name in its settings. You can then switch to the easy screen and check the user interface that you have created. Once you are satisfied with it, save it as a normal preset while you are on the easy screen. Although your preset will not be displayed or selected in the list of available devices, the functionality will be exactly the same. For more information about multiparameters and devices please check the **online video tutorials**.

If you are an advanced designer, you can also view both the easy and edit screens at the same time. To do that, hold **Ctrl** key and press the Edit button.

# Edit mode



## Global presets

### 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.



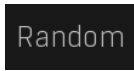
### Right arrow

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.

## Multi-core

### MultiCore

MultiCore switch lets the plugin exploit the full potential of your CPU by parallelizing rendering of audio for different voices. Note that in certain cases this may actually lower the performance depending on the host and other plugins. The performance can be especially degraded when small processing buffers hence small latency is used. On the other hand the higher number of voices is used, the bigger the gain from the parallelization can be.

## Settings

### 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.

## Sleeping

### Sleep indicator

Sleep indicator informs whether the plugin is currently active or in sleep mode. The plugin can automatically switch itself off to save CPU, when there is no input signal and the plugin knows it cannot produce any signal on its own and it generally makes sense. You can disable this in Settings / **Intelligent sleep on silence** both for individual instances and globally for all plugins on the system.

# Plugin toolbar



Plugin toolbar provides some global features, A-H presets and more.

1x

### 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.

Set

### Set

Set button uses the AGC (automatic gain compensation) processor to calculate the ideal output gain to ensure that the output audio loudness is equal to the input level. To use it, simply enable playback in your host and click the button. The plugin's output gain will be adjusted to match the input and output levels as closely as possible.

If the AGC is already enabled, the change will be instant and you can disable the AGC afterwards. Typically you will browse presets, generate random settings etc. During the entire time you will have AGC enabled to prevent you from experiencing different output loudness levels. When you find a sonically ideal setup, you simply click the Set button to set the output gain automatically and disable the AGC as you won't need it anymore.

If the AGC is not already enabled, clicking the Set button displays a window with progress bar for a few seconds, while the plugin temporarily enables AGC and analyses input and output of the plugin. After that the AGC is disabled again.

To get the best results, you should feed the plugin with some "universal" signal. If you are processing a specific instrument, play a typical part, a chorus in case of vocals for example. If you are creating presets designed for general use, white/pink noise may be the best signal to use.

Limiter

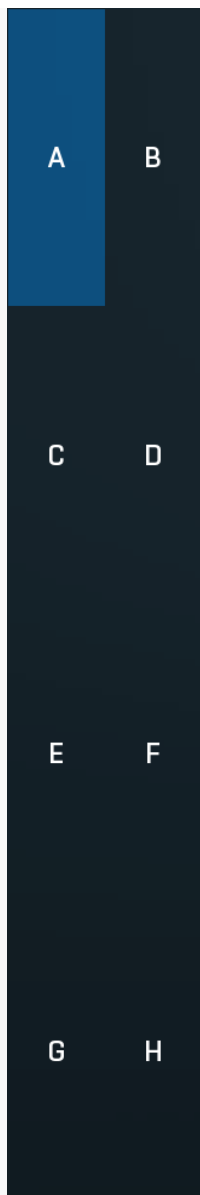
### Limiter

Limiter button enables or disables the safety limiter. Its purpose is to protect you from peaks above 0dB, which can have damaging effects to your processing chain, your monitors and even your hearing.

It is generally advised to keep your audio below 0dB at all times in all stages of your processing chain. However, several plugins may cause high level outputs with certain settings, often due to unprevented resonances with specific audio materials. The safety limiter prevents that.

Note that it is NOT wise to enable this "just in case". As with any processing, the limiter requires additional processing power and modifies the output signal. It is a transparent single-band brickwall limiter, but you still need to be careful when using it.





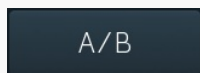
### A-H presets selector

A-H presets selector controls the current A-H preset. This allows the plugin to store up to 8 sets of settings, including those parameters that cannot be automated or modulated. However it does not include channel mode, oversampling and potentially some other global controls available from the Settings/Settings menu.

For example, this feature can be used to keep multiple settings, when you are not sure about the ideal configuration. When you change any parameter, only the currently selected preset is modified.

The four buttons below enable you to switch between the last 2 selected sets using the A/B button, morph between the first 4 sets using the morphing button and copy & paste settings from one preset to another (via the clipboard).

It is also possible to switch between the presets using MIDI program change messages sent from your host. The set selected depends on the Program Change number: 0 selects A, 7 selects H, 8 selects A, 15 selects H and so on.



### A/B

A/B button switches between the active and previously active A-H preset (not necessarily the A and B presets themselves). To compare any 2 of the A-H presets, select one and then the other. Clicking this button will then switch between these two. You can do the same thing by clicking on the particular presets, but this makes it easier, letting you close your eyes and just listen.



### Morph

Morph button lets you morph between the A, B, C and D settings. Morphing only affects those parameters that can be automated or modulated; that does include most of the parameters however. When you click this button, an X/Y graph is shown allowing you to drag the position indicator to any position between the letters A, B, C and D. The closer you drag the indicator to one of the letters, the closer the actual settings are to that preset.

**Please note that this will overwrite and change the preset that is currently selected, so it is best to select a new preset e.g. 'E', then use the morphing method. This way you will define the settings for A, B, C and D, morph between them, and store the result in 'E' without any modification of the original A, B, C and D presets.**

Please note that the ABCD morphing itself cannot be automated and that, while morphing, the changes to the underlying parameters

are not notified to the host (there may be hundreds of change events).



### Copy

Copy button copies the current settings to the system clipboard. Other presets, oversampling, channel mode and other global settings are not copied.

Hold **Ctrl** to save the settings as a file instead. That may be necessary for complex settings, which may be too long for system clipboard to handle. It may also be advantageous when you want to send the settings via email. You can load the settings by drag & dropping them to a plugin or holding **Ctrl** and clicking **Paste**.



### Paste

Paste button pastes settings from the system clipboard into the current preset. Hold **Ctrl** to load the settings from a file instead. Hold **Shift** to paste the settings to all of the A-H slots at once.



### Undo

Undo button reverts the last change. Only changes to automatable or modulatable parameters and global settings (load/randomize) are stored.



### Redo

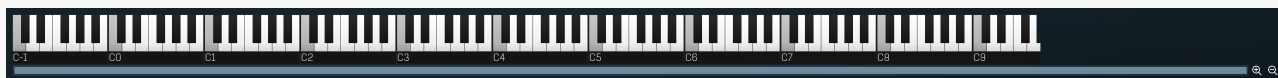
Redo button reverts the last undo operation.



### Collapse

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

## Piano view

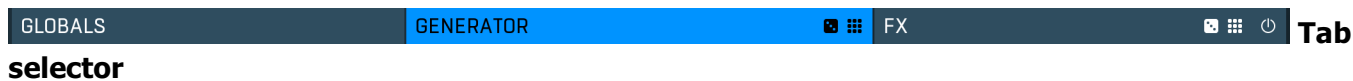


Piano view lets you play and display MIDI notes



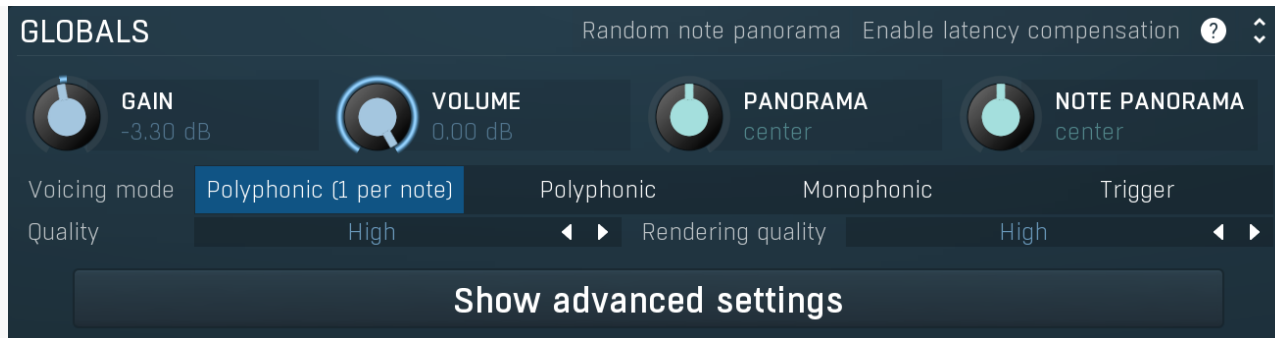
### Collapse

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



Tab selector switches between subsections.

## Globals panel



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

### Random note panorama

#### Random note panorama

Random note panorama switch changes the behaviour of **Note panorama**. Normally that parameter sets the panorama directly for each note, so you can for example use a modulator to control it. When this switch is enabled, Note panorama is merely a range and the initial panorama of each note will be randomly selected from the range -Note panorama to +Note panorama. Hence it also doesn't matter if the current Note panorama is right or left. The actual panorama of the sound is determined by Note panorama at the point of the note being pressed and by Panorama while the sound is generated.

### Enable latency compensation

#### Enable latency compensation

Enable latency compensation switch enables latency compensation for FX modules. Note that if you use such modules and enable this option, the plugin won't be usable live, since there would be a delay. With this option disabled you can still play live if the latency introducing modules are on a parallel lane. That can still be useful for instance to process delays of the original sound, where the exact timing may not matter.



#### Gain

Gain defines the output gain. It is applied after all the generators and effects are processed.

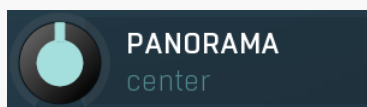
Range: -48.00 dB to +48.00 dB, default 0.00 dB



#### 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 processed.

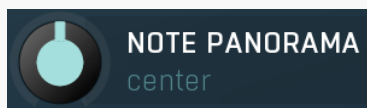
Range: silence to 0.00 dB, default 0.00 dB



#### Panorama

Panorama defines the output panorama. It is applied after all the generators and effects are processed.

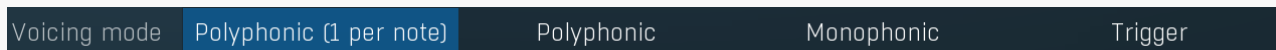
Range: 100% left to 100% right, default center



#### 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



#### Voicing mode

Voicing mode defines the way in which the instrument works when multiple notes are played.

**Polyphonic** mode is the default mode, which enables the instrument to create as many simultaneous voices as necessary, up to a limit of 256, after that the earliest voice is dropped.

**Polyphonic (single voice per note)** is similar to polyphonic, but ensures that only one voice will be created at any time for each MIDI key played. 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 the sustain pedal, however it may cause unexpected results when harmonies are used, as pressing one key simulates pressing multiple keys.

**Monophonic** mode makes the instrument 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.

**Trigger** mode makes the instrument completely ignore note release events (Note Offs) and makes the notes be triggered on and off immediately providing a sort of staccato behaviour. Effectively it uses the main ADSR envelope shape, without holding at the Sustain stage (the other modes do hold at this stage). Accordingly, the sounds can still be shaped using this envelope. This is especially useful for percussive sounds and when used in conjunction with another instrument to form special attack sounds to accompany the sound

produced by that other instrument, emphasising transients, for example.

## Quality High 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.

## Rendering quality High 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.

### Show advanced settings

#### Advanced settings

Advanced settings button displays additional settings.

## Global pitch panel



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

### Octaves 0 Octaves

Octaves defines the global pitch change in octaves.

Range: -8 to +8, default 0

### Semitones 0 Semitones

Semitones defines the global pitch change in semitones.

Range: -24.00 to +24.00, default 0

### Pitch-bend range 48.00 Pitch-bend range

Pitch-bend range controls the range in semitones up or down of the global pitch change caused by the MIDI pitch-bend controller. For example, a value of 48 is a range of +/-48 semitones (that is +/- 2 octaves). This is especially useful with MPE controllers, which tend to have a predefined range letting the users slide through the entire range of the controller.

Range: 0.00 to 48.00, default 48.00

### Tuning 440.0 Hz Tuning

Tuning controls the global tuning, also called pitch standard, being the frequency of A above middle C.

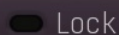
Range: 400.0 Hz to 480.0 Hz, default 440.0 Hz

Default tuning

### Default tuning

Default tuning button restores the default equal temperament tuning.

Load tuning



Lock

### 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).



Lock

### Lock tuning

Lock tuning switch makes the engine keep the microtuning when browsing presets.

## Analog panel



Analog panel controls parameters for simulating inaccuracies found in analog devices. To mimic a true analog synth, you need 3 essential elements :

1. Nonlinearities, which you can introduce using various distortion modules such as Saturator.
2. Noise, which you can generate using NoiseGenerator module.
3. Introduce inaccuracies and make various parameters move in time. Especially frequencies of oscillators, filters etc. This is the key element, which makes every voice slightly different giving it the lush 'alive' character. It can be emulated using the Random per-voice modulation, but using the analog drift is much easier. This panel is the master control for all parameters this applies to - all parameters, which can be modulated. Each of these parameters has its own analog drift control in its modulation matrix so you can also control how much each parameter is affected. Many of these controls (e.g. detuning, frequencies, envelope times...) already have set the default analog drift, so in most cases you can simply enable this feature and you are done.



### Depth

Depth controls the total amount of analog drifting for all affected parameters. Note that we intentionally made the range very high, from no drifting at all to extreme movement useful for sound effects and experiments.

Range: 0.00% to 100.0%, default 10.0%

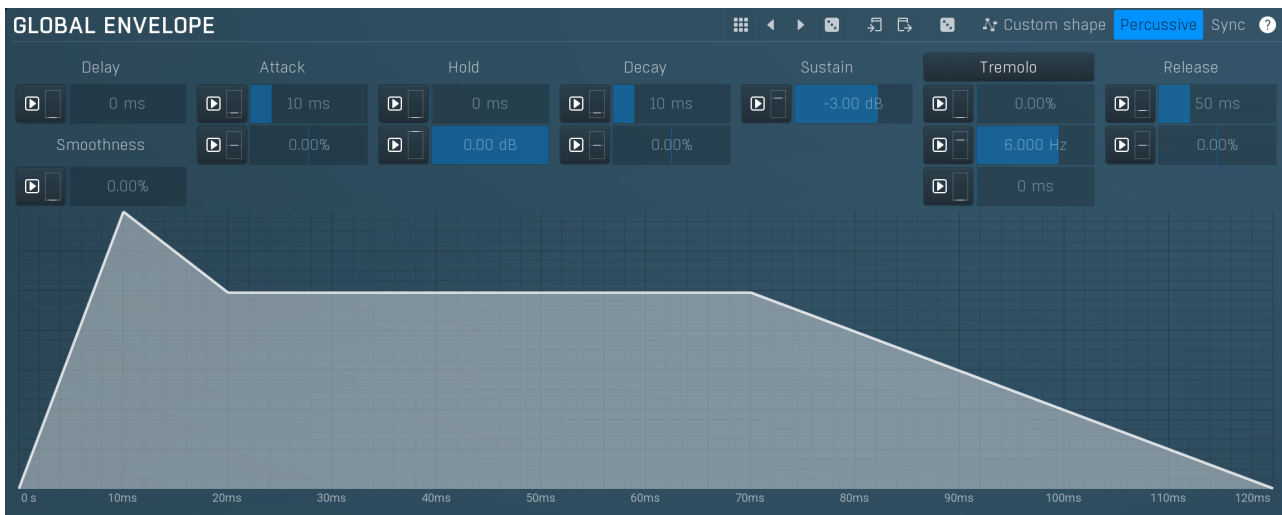


### Speed

Speed controls the speed of the analog drift, hence how quickly are the parameters randomly changing.

Range: -100.0% to 100.0%, default 0.00%





**Global**

## envelope graph

Global envelope graph controls the global 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.

# Envelope graph menu



Envelope graph menu provides additional features which are used to edit the graph. Open the menu using right mouse button in the graph. Please note that if you select some points in the graph, or click on a point for example, the menu will be different and will cover only those features related to the selected set of 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.



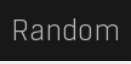
### Copy

Copy button copies the settings onto the system clipboard.



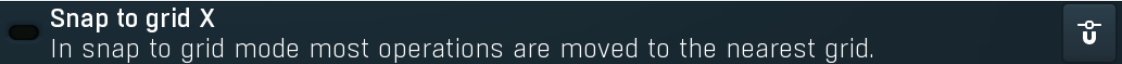
### Paste

Paste button loads the settings from the system clipboard.



### Random

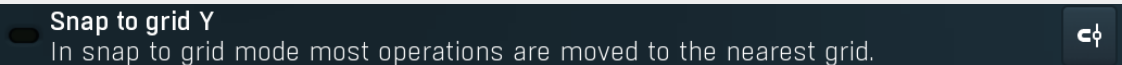
Random button generates random settings using the existing presets.



### Snap to grid

X

Snap to grid X activates the snap to grid feature. Alternatively you can press **Alt** while dragging a point or selection.



### Snap

Snap button activates the snap to grid feature. Alternatively you can press **Alt** while dragging a point or selection.



### Insert point

Insert point button creates a point at mouse position.



### Step sequencer

Step sequencer button generates the envelope from step sequencer.

## Clear points

## Clear points

Clear points button deletes all points.

## Distribute points

## Distribute points

Distribute points button makes all points equally spaced.

## Randomize

## Randomize

Randomize button slightly modifies the Y coordinates.

## Mirror X

## Mirror X

Mirror X button inverts the X coordinates of all points.

## Mirror Y

## Mirror Y

Mirror Y button inverts the Y coordinates of all points.

## Export CSV

## Export CSV

Export CSV feature lets you export the graph to a CSV file. CSV file is a simple text format, which has multiple lines with X and Y coordinates delimited by ';'. For example:

```
0.275;0.2
0.438;0.5
0.775;0.67
```

## Import CSV

## Import CSV

Import CSV feature lets you select a CSV file and imports the graph points from it. CSV file is a simple text format, which has multiple lines with X and Y coordinates delimited by ';'. For example:

```
0.275;0.2
0.438;0.5
0.775;0.67
```

## Expression evaluator

## Expression evaluator

Expression evaluator lets you generate points based on a mathematic formula. The only input variable is 'x', so as an example you may write 'ln(x^3 + 1) - sin(x\*x)'.

Expression evaluator uses traditional C/C++ style formatting, which is natural for most people. It provides arithmetics, logical and conditional operators. Following terms are supported:

Constants: **pi**, **e**, **sqrt2**, **ln2**

Arithmetic operators:

**-a** inverts the sign, e.g. "-x" produces +2 for x=-2

**a+b** = addition

**a-b** = subtraction

**a\*b** = multiplication

**a/b** = division

**a%b** = modulo, remainder after division

**a^b** = power, e.g. "2^3" produces 2\*2\*2 = 8

Arithmetic functions:

**min(a,b)** = minimum of both values

**max(a,b)** = maximum of both values

**limit(a,min,max)** = a limited into the interval min..max

**to01(a,min,max)** = converts "a" as min..max to 0..1

**from01(a,min,max)** = converts "a" as 0..1 to min..max

**tom11(a,min,max)** = converts "a" as min..max to -1..1

**fromm11(a,min,max)** = converts "a" as -1..1 to min..max

Basic mathematic functions: **abs(x)** = absolute value, e.g. abs(-3) = 3 **sqr(x)** = x\*x **sqrt(x)** = square root **exp(x)** = natural exponential **e^x** **ln(x)** = natural logarithm **log10(x)** = logarithm with base 10 **log(x, base)** = logarithm with specified base **inv(x)** = 1/x **sgn(x)** = sign of x, -1 or 0 or +1 depending on x\*x **round(x)** = rounding to the nearest value **floor(x)** = rounding to the nearest lower value, e.g. floor(-2.3) = -3 **ceil(x)** = rounding to the nearest higher value, e.g. ceil(-2.3) = -2 **rand(x)** = random value from 0 to x

Functions for specific units:

**f01(a)** = converts "a" as frequency from 20...20000 into log scale 0..1

**ffrom01(a)** = converts "a" as 0..1 (log scale) to frequency from 20...20000

**todb(a)** = converts "a" as multiplier to dB value by calculating "20\*log10(a)"

**fromdb(a)** = converts "a" as dB value to multiplier by calculating "10^(a/20)"

Trigonometric functions:  **$\sin(x)$** ,  **$\text{asin}(x)$** ,  **$\cos(x)$** ,  **$\text{acos}(x)$** ,  **$\tan(x)$** ,  **$\text{atan}(x)$** ,  **$\sinh(x)$** ,  **$\cosh(x)$** ,  **$\tanh(x)$**

Logical operators:

**$a==b$**  = comparison producing 1 if "a" and "b" are equal, 0 otherwise

**$a!=b$**  = comparison producing 1 if "a" and "b" are NOT equal, 0 otherwise

**$a<b$**  = comparison producing 1 if "a" is lower than "b", 0 otherwise

**$a<=b$**  = comparison producing 1 if "a" is lower or equal to "b", 0 otherwise

**$a>b$**  = comparison producing 1 if "a" is greater than "b", 0 otherwise

**$a>=b$**  = comparison producing 1 if "a" is greater or equal to "b", 0 otherwise

**$!a$**  = logical negation, 0 produces 1, 0 otherwise

**$a\&\&b$**  = logical AND, produces 1 if both "a" and "b" are nonzero

**$a\|\b$**  = logical OR, produces 1 if any of "a" and "b" are nonzero

**$a\wedge\wedge b$**  = logical XOR, produces 1 if "a" and "b" are logically different

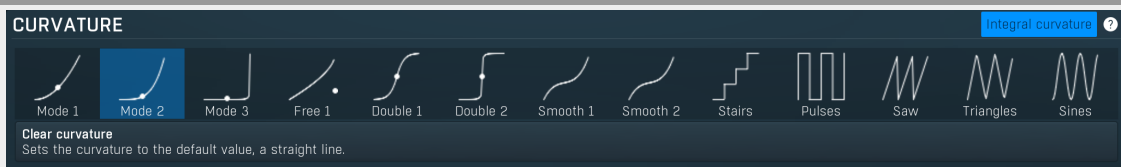
**$a?b:c$**  = if a is nonzero, then the result is b, otherwise it is c

Analyse audio

## Analyse audio

Analyse audio lets you analyse a portion of an audio file at specified intervals, extract its level envelope and use those levels to construct the graph's curve.

## Curvature

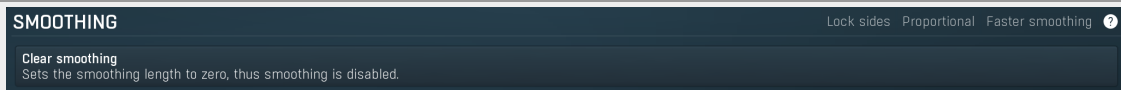


Integral curvature

## Integral curvature

Integral curvature makes the multi-curvature modes such as rectangles always have an integral number of items, e.g. 1, 2, 3, ... rectangles. If you disable this, it will be also possible to have for example 2.3 rectangles, which will however cause a discontinuity.

## Smoothing



Lock sides

## Lock sides

Lock sides makes the smoothing factor equal on both sides.

Proportional

## Proportional

Proportional makes the smoothing area size defined by the smaller side.

Faster smoothing

## Faster smoothing

Faster smoothing enables slightly faster algorithm, which can however often cause unnecessary curving.

Tremolo

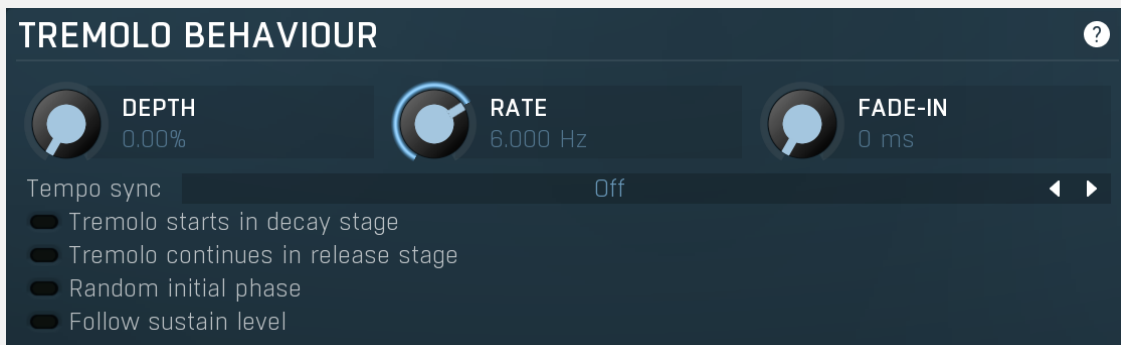
## Tremolo

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.





## sync

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

### Tremolo starts in decay stage

## 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

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

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

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

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

Random button generates random settings using the existing presets.

## Normal

## 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.

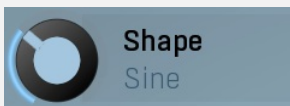
## Harmonics

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

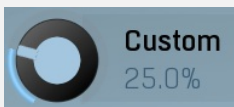


Signal generator in Normal mode works by generating the oscillator shape using a combination of several curves - a predefined set of standard curves, custom shape, step sequencer and custom sample. It also post-processes the shape using several filters including smoothing to custom transformations. This 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.



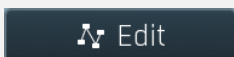
#### Shape

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

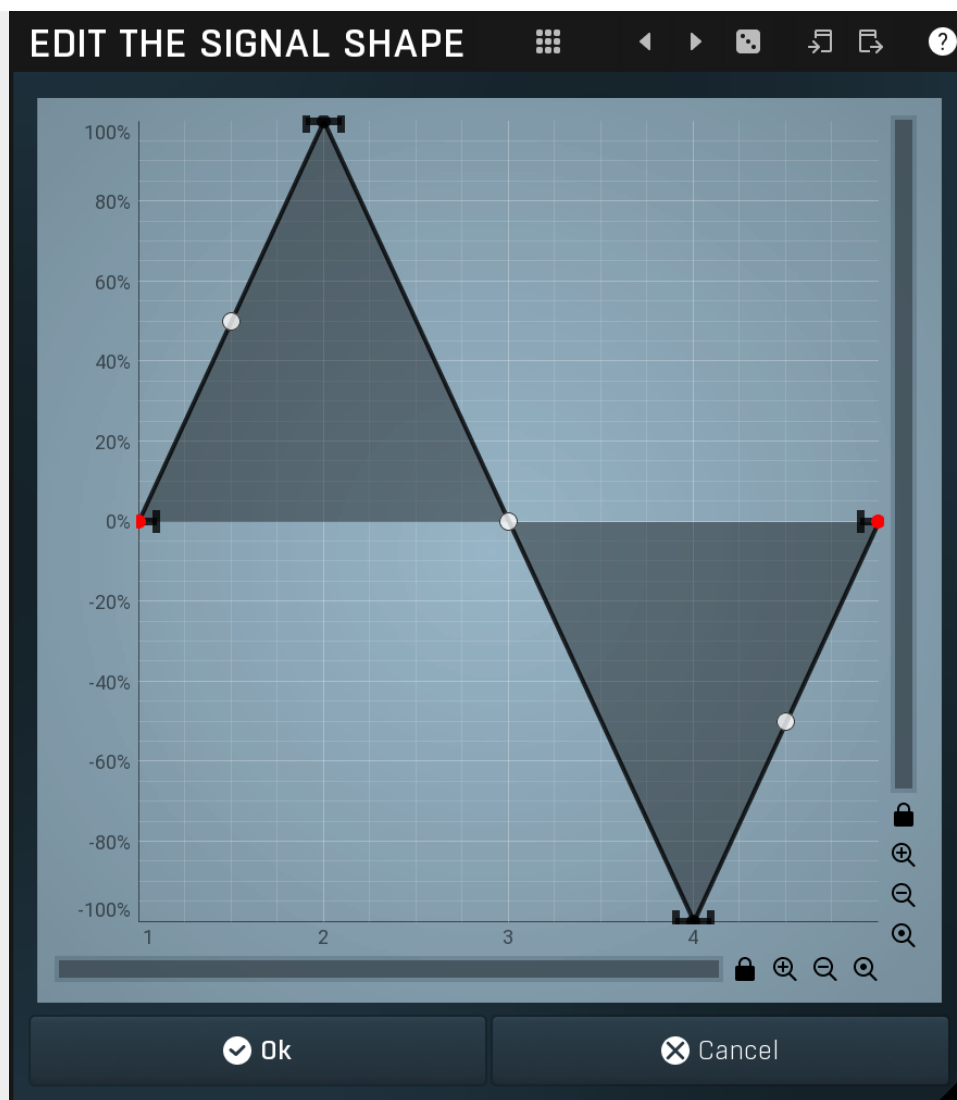
Custom controls the amount of the custom shape that is blended into the main shape.



#### Edit

Edit button shows the custom shape editor.

### Signal generator custom shape editor



Signal generator custom shape editor controls the custom shape. You can edit virtually any shape that you can imagine and then blend it with the standard shapes, the step sequencer 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

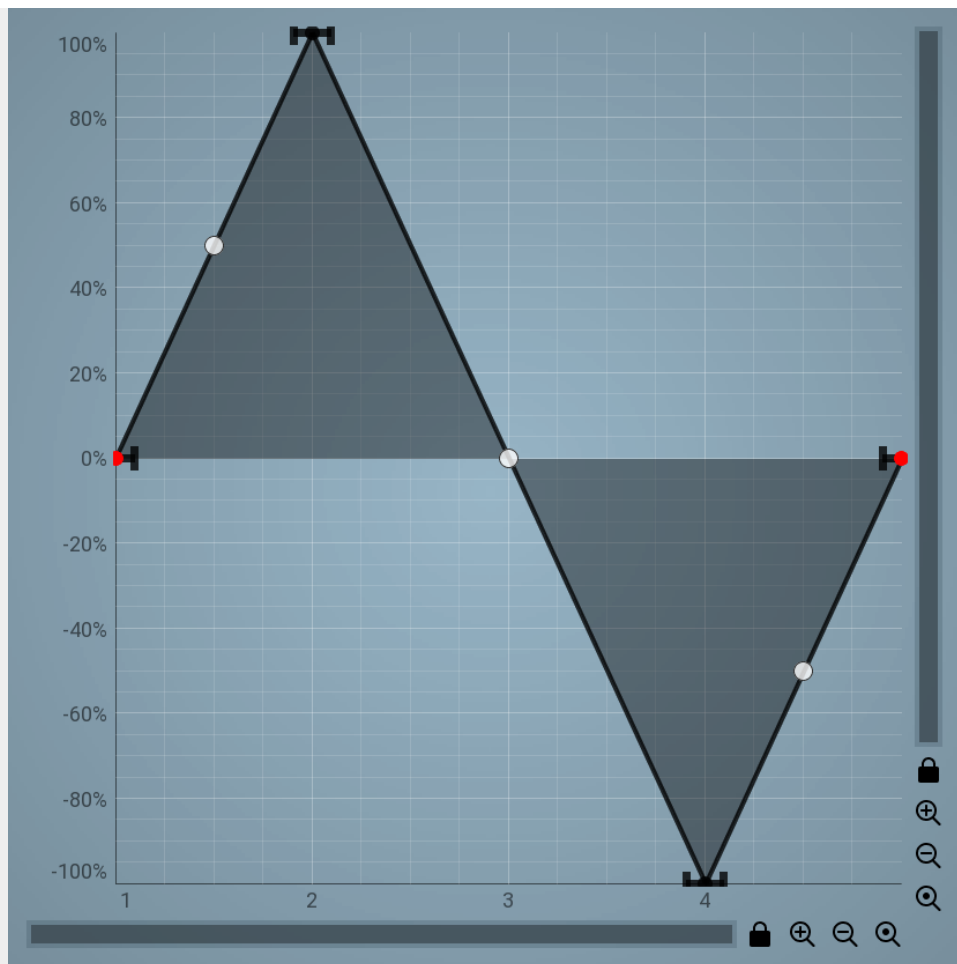
Randomize button loads a random preset.

### Copy

Copy button copies the settings onto the system clipboard.

### Paste

Paste button loads the settings from the system clipboard.



**Graph editor**

Graph editor lets you edit the envelope graph.

## 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.



### Step

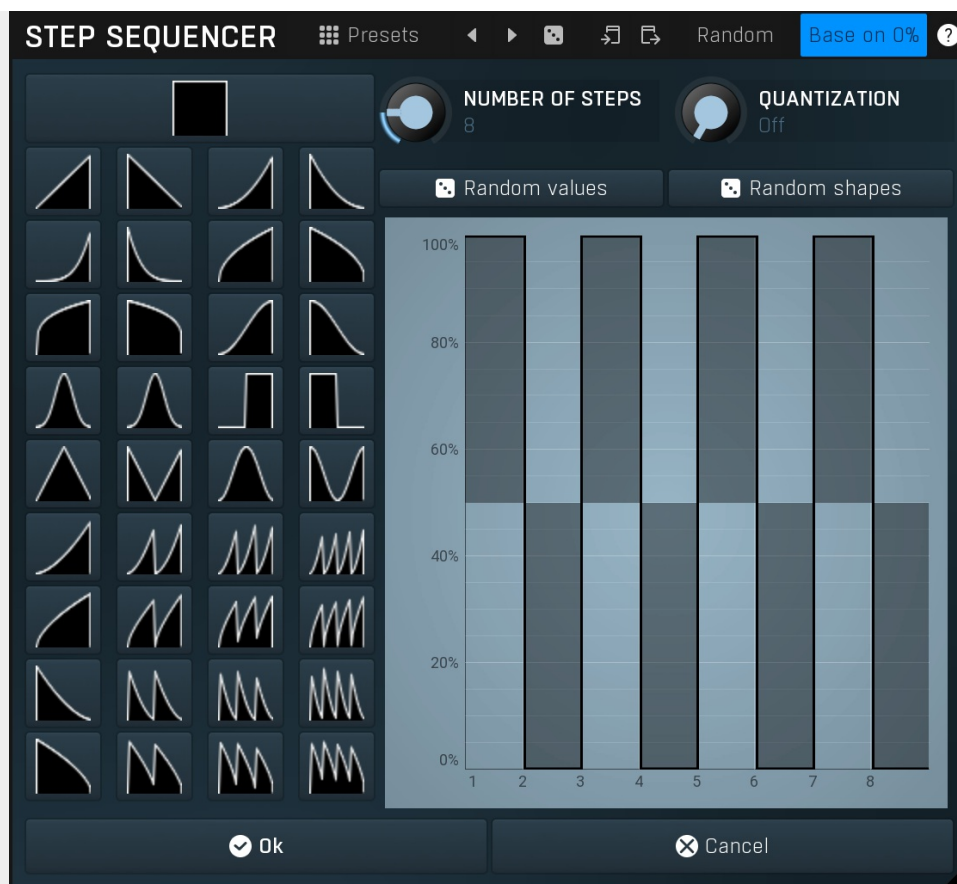
Step controls the amount of the step sequencer shape that is blended into the main shape (which has already been blended with the custom shape).



### Edit

Edit button shows the step sequencer editor.

## Signal generator step sequencer editor



Signal generator step sequencer editor controls the step sequencer shape. You can have various numbers of steps each with a different value and shape. Note that for classic rectangular shapes the output can be very rough, hence it may be worth considering using **Smoothness** parameter to smooth out the resulting shape. This will use additional CPU power of course, but that should be negligible unless you modulate any of the signal generator parameters.



### 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.



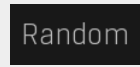
### Copy

Copy button copies the settings onto the system clipboard.



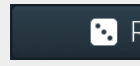
### Paste

Paste button loads the settings from the system clipboard.



### Random

Random button generates random settings using the existing presets.



### Random values

### Random values

Random values button generates random sequence of values, but keeps the shape of each step.

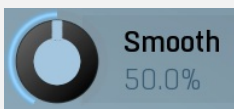


### Random shapes

### Random shapes

Random shapes button generates random sequence of shapes, but keeps the values of each step.





Smooth  
50.0%

## Smooth

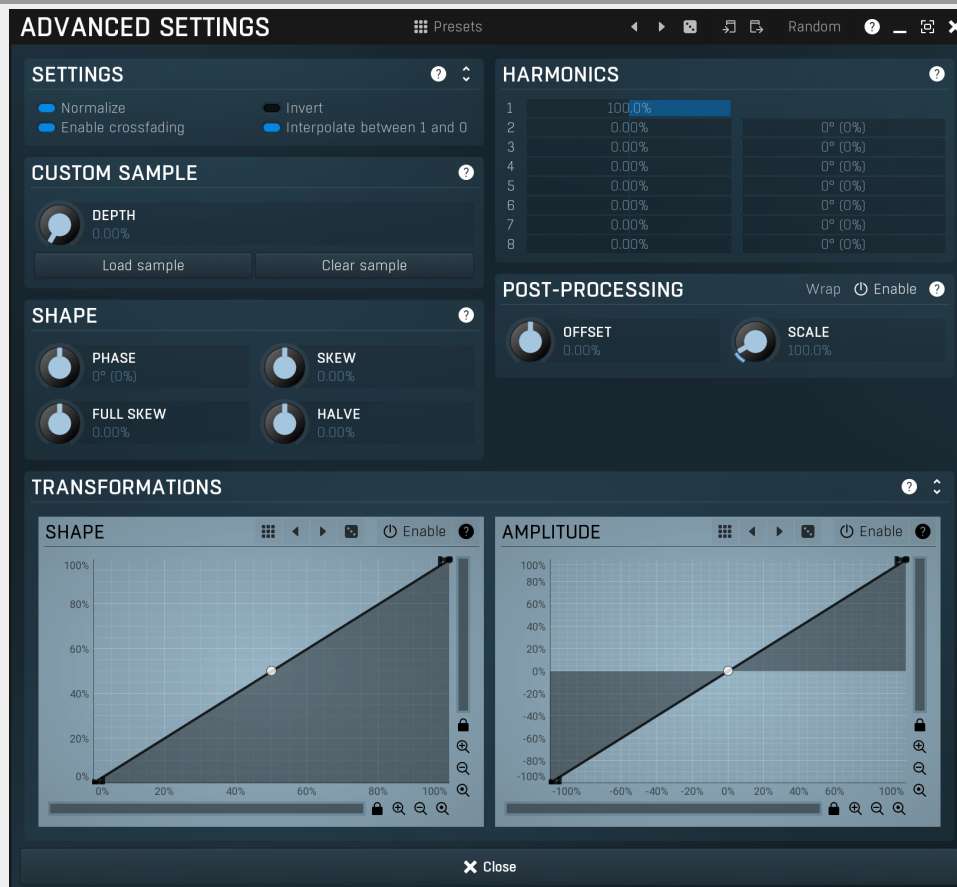
Smooth controls the amount of smoothing. Many shapes, especially those produced by the step sequencer, have rough jagged edges, which may be advantageous, but when used to modulate certain parameters, the output may be clicking or causing other artifacts. Smoothness helps it by smoothing the whole signal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

Advanced

## Advanced

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

## Advanced settings



Presets

## 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.



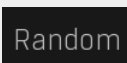
### Copy

Copy button copies the settings onto the system clipboard.



## Paste

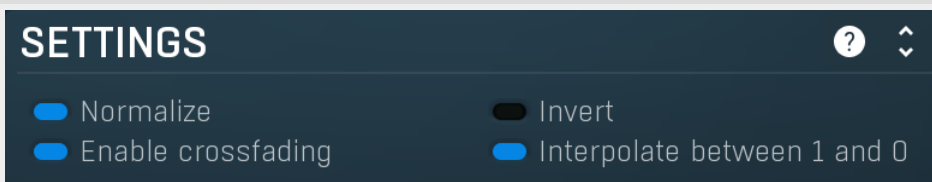
Paste button loads the settings from the system clipboard.



## Random

Random button generates random settings using the existing presets.

## Settings panel



Settings panel contains some global settings of the oscillator.



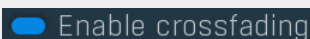
### Normalize

Normalize switch enables normalization to -1..+1. It is generally desirable since even if you draw a custom shape, you usually want it to have the full range. You may want to disable it if you want to create some custom shapes, where the level actually matters.



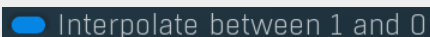
### Invert

Invert switch simply inverts the output shape vertically.



### Enable crossfading

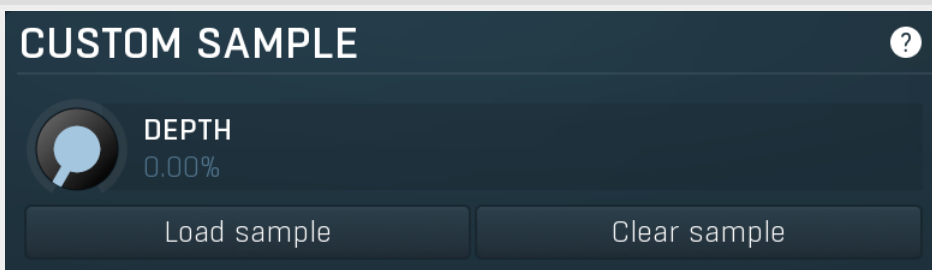
Enable crossfading enables interpolation between shapes when the shape is changing. This requires more CPU, but can avoid zipper noise when the shape is being modulated for example.



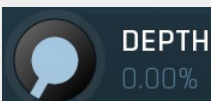
### Interpolate between 1 and 0

Interpolate between 1 and 0 smoothens the discontinuity between 1 and 0 values, which is inevitable for shapes such as saw or rect for example. However when this is a high frequency oscillator (HFO), this discontinuity is what creates the highest frequencies, so it is actually desirable. When using it as an LFO, you may also want the discontinuity in some extreme cases.

## Custom sample panel



Custom sample panel contains parameters of the custom sample that you can load and mix with the other sources. Do NOT confuse this with a sampler, the custom sample is taken as one period of the waveform. It can be used for creative effects and it can be used to import a custom waveform. The custom sample is then stored with limited precision within the settings, so the sample does not need to be kept on the system, but note that these settings may be quite large. To limit the space required by the settings, the sample is stored only if the depth is not 0%, meaning only if the sample is actually used.



### Depth

Depth controls the amount of custom sample mix. 0% means the sample is not used even if there actually is one loaded. 100% means the sample completely overrides the basic shape, custom shape, step sequencer... However, transformations are still performed on the sample.

Load sample

### Load sample

Load sample button displays a file selection window, which lets you select the custom sample file.

Clear sample

### Clear sample

Clear sample button removes the custom sample if it has been loaded.

## Shape panel

## SHAPE



PHASE

0° (0%)



SKEW

0.00%



FULL SKEW

0.00%



HALVE

0.00%

Shape panel contains parameters performing various transformations on the signal shape. Please note that most transformation require a significant amount of CPU resources, so you should not automate or modulate the signal shape if you are using them.

## Harmonics panel

### HARMONICS



1	100.0%	
2	0.00%	0° (0%)
3	0.00%	0° (0%)
4	0.00%	0° (0%)
5	0.00%	0° (0%)
6	0.00%	0° (0%)
7	0.00%	0° (0%)
8	0.00%	0° (0%)

Harmonics panel lets you add separate harmonics of the original signal.

## Post-processing panel

### POST-PROCESSING

Wrap Enable



OFFSET

0.00%

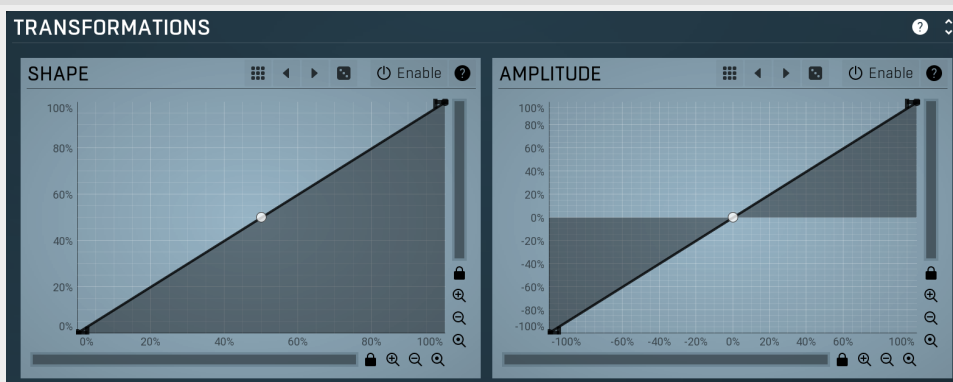


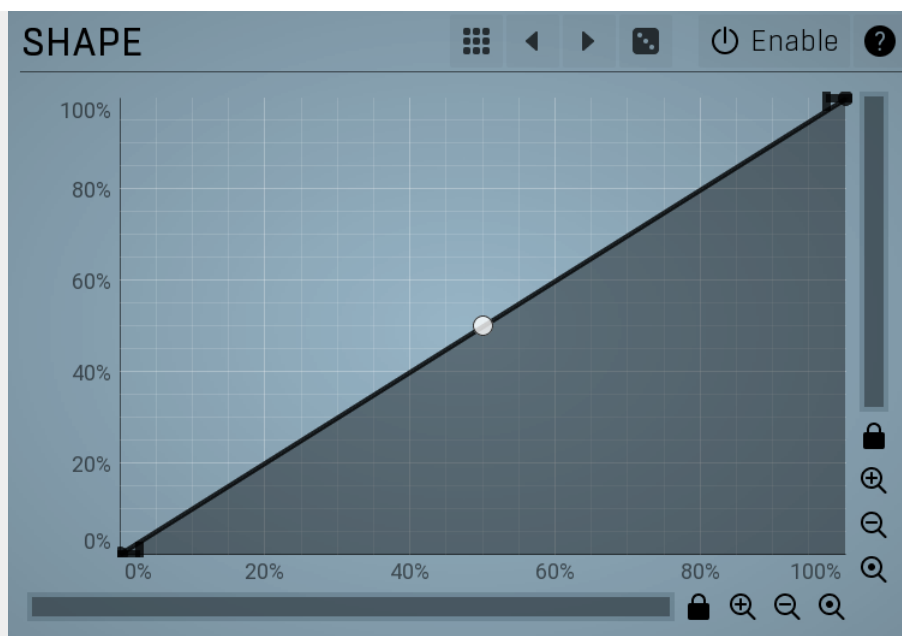
SCALE

100.0%

Post-processing panel lets you post-process the shape after all the previous generator items.

## Transformations





**Shape**

### transformation graph

Shape transformation graph lets you perform arbitrary modification of the graph shape. Basically this graph lets you modify the shape "in time". The Y axis represents the position in the source signal related to the position in the target signal. The best way to check what it does is simply to try it.

#### 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

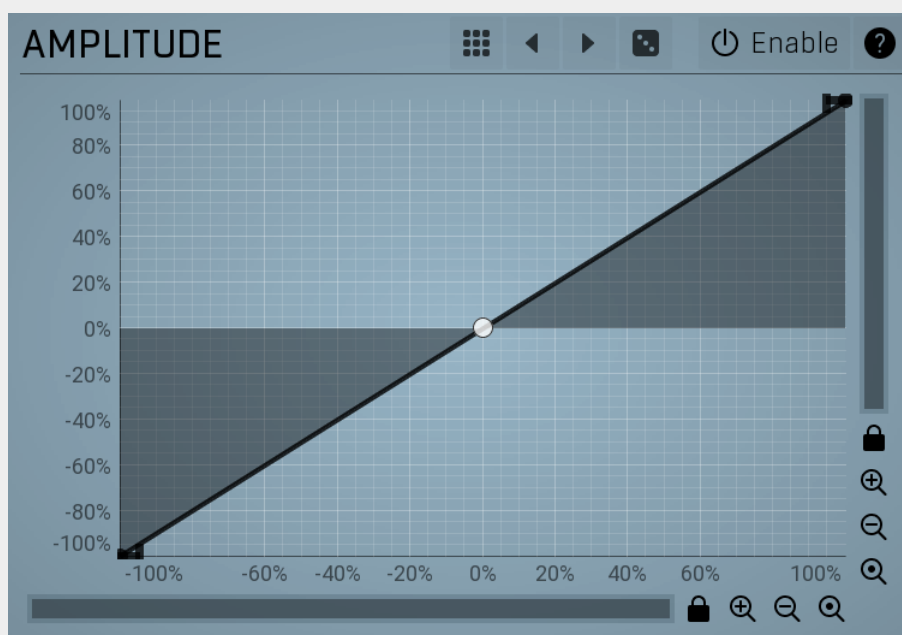
Left arrow button loads the previous preset.

#### Right arrow

Right arrow button loads the next preset.

#### Randomize

Randomize button loads a random preset.



**Amplitude**

### transformation graph

Amplitude transformation graph lets you perform arbitrary modification of the graph amplitude. Basically this graph lets you modify the shape's level, vertical axis. The X axis represents the original values, the Y axis defines the resulting values. The best way to check what it does is simply to try it.

#### 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

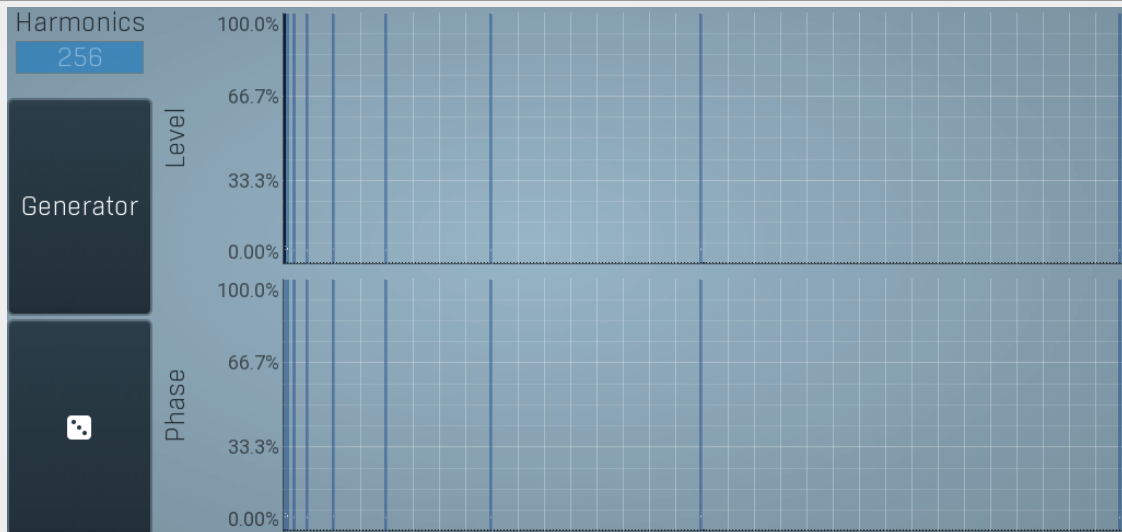
Right arrow button loads the next preset.



### Randomize

Randomize button loads a random preset.

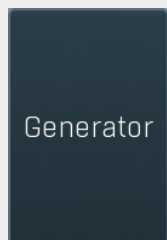
## Signal generator in Harmonics mode



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.

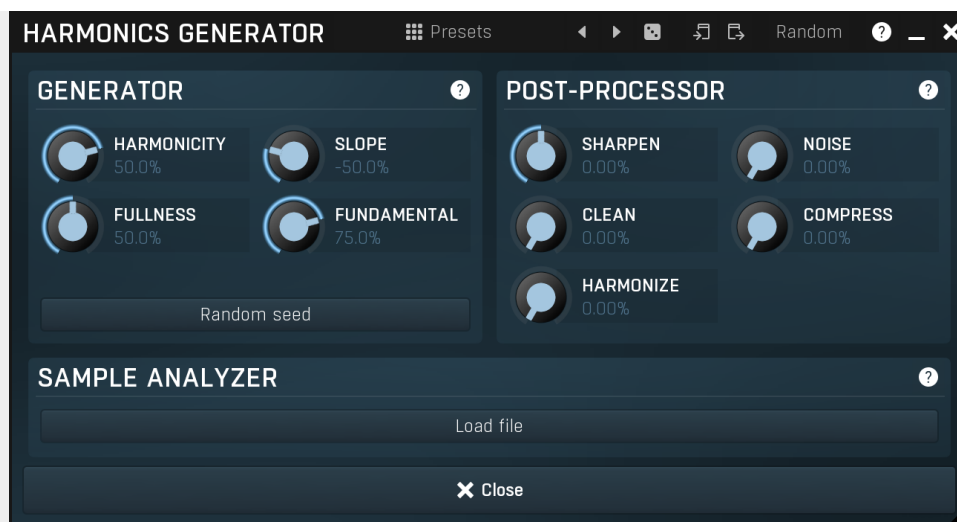
Editing harmonics can be time consuming unless you hear what you want, so a signal generator is also available. This great tool lets you generate a random spectrum by a single click. You can also open the **Generator** settings and edit its parameters, which basically control the audio properties in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

## Harmonics generator



Harmonics generator is a powerful tool, that can generate various harmonics-based timbres and even analyze a sample file and extract harmonics from it.

 Presets

## 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

Random button generates random settings using the existing presets.

## Generator panel



Generator panel contains parameters of the harmonics generator. By changing any of the parameters, the harmonics are changed, however only **Random seed** button changes the structure completely. The other parameters can be used to tweak the results.



## HARMONICITY

50.0%

### Harmonicity

Harmonicity controls the ratio between natural harmonics and those which sound disharmonic (despite the title "harmonics"). Assuming that the 1st harmonic is the fundamental, 2nd harmonic is 1 octave above, 4th is 2 octaves above, both can be considered very natural. 3rd harmonic is 1 octave and a 5th above the fundamental, and is still pretty harmonic, but less than the octaves. 5th harmonic is 2 octaves and a major 3rd above the fundamental. Such a tone may sound very disharmonic, in minor scales for example. Higher harmonics are often very disharmonic and produce typical ringing timbres.

When harmonicity parameter is set to 100%, only octaves are allowed. By lowering the value more and more disharmonics are created and with 0% all frequencies are allowed. For values below 0% disharmonics are preferred, hence you can expect more ringing timbres.

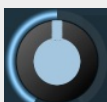


## SLOPE

-50.0%

### Slope

Slope defines the amount of higher harmonics compared to lower ones. When 0%, the higher harmonics have the same levels as lower ones. Typically you use values below 0%, which attenuates the higher harmonics making the resulting sound darker. Similarly values above 0% make the sound brighter.



## FULLNESS

50.0%

### Fullness

Fullness controls the number of generated harmonics. With values around 0% the resulting timbres will contain only a few harmonics making the sound clear. Higher values increase number of harmonics making the timbre rich.



## FUNDAMENTAL

75.0%

### Fundamental

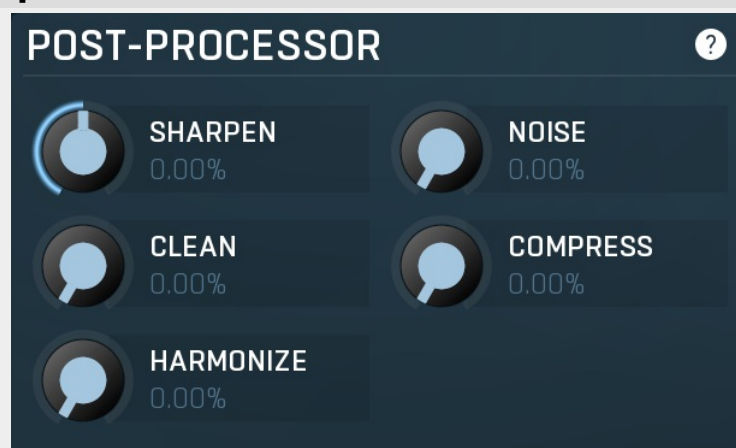
Fundamental controls the minimum level of the fundamental (the 1st harmonic). Most sounds have a very strong fundamental as it carries the pitch.

Random seed

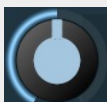
### Random seed

Random seed button generates a new series of harmonics. Pressing this button will create a whole new timbre.

## Post-processor panel



Post-processor panel contains parameters of the harmonics post-processor. The generator and sample analyzer first create a series of harmonics, the timbre. These harmonics are mixed depending on the **Sample ratio** parameter. After that the post-processor is engaged, which can further transform the harmonics in several ways.



## SHARPEN

0.00%

### Sharpen

Sharpen is a sort of soft compression/expanding unit. Values below 0% decrease the level of quiet harmonics, while values above 0% increase their level.



## NOISE

0.00%

### Noise

Noise defines amount of noise added to the timbre. Noise can make the results dirty providing much richer timbres.





### Clean

Clean controls the threshold of a gate. It basically attenuates or removes harmonics below this level making the output cleaner.



### Compress

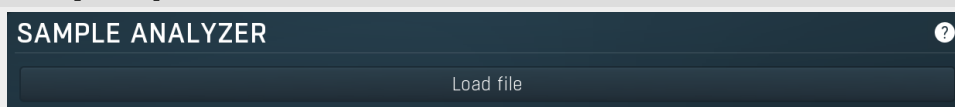
Compress reduces the dynamic range of the harmonics by increasing levels of the quiet ones, but keeping the levels of the loud ones.



### Harmonize

Harmonize creates additional higher harmonics from existing ones. This is especially useful to transform rich dirty disharmonic timbres into similarly rich but more harmonic timbres.

## Sample analyzer panel

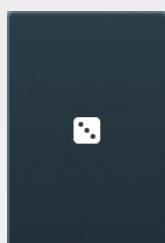


Sample analyzer panel contains parameters of the sample analyzer. If there is no sample loaded, the sample analyzer is turned off. The analyzer takes the selected sample and a position within it, analyses one period of the signal waveform and produces the output set of harmonics. You can then combine these harmonics with the output of the generator using **Sample ratio** parameter.

The sample itself is not store with the plugin settings. Instead the path to the target sample file is stored along with the analyzed harmonics. If the sample file is not available, you cannot modify the analysis parameters and the last analyzed harmonics are used. This means that you actually don't need to have the sample file available on the computer on which you are using the settings.

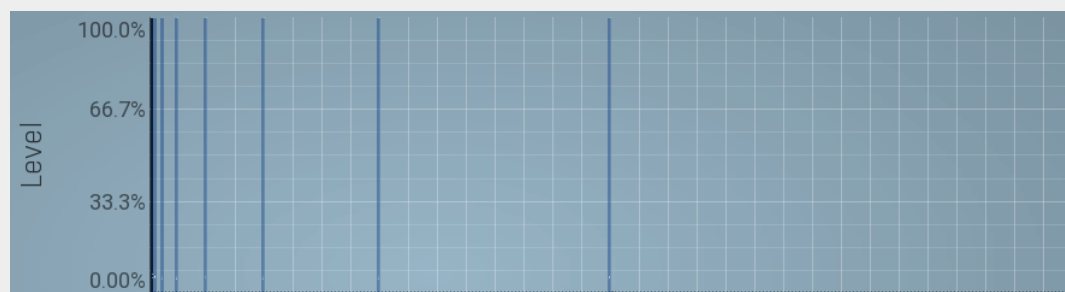


Load file button lets you select a sample file to analyse.



### Randomize

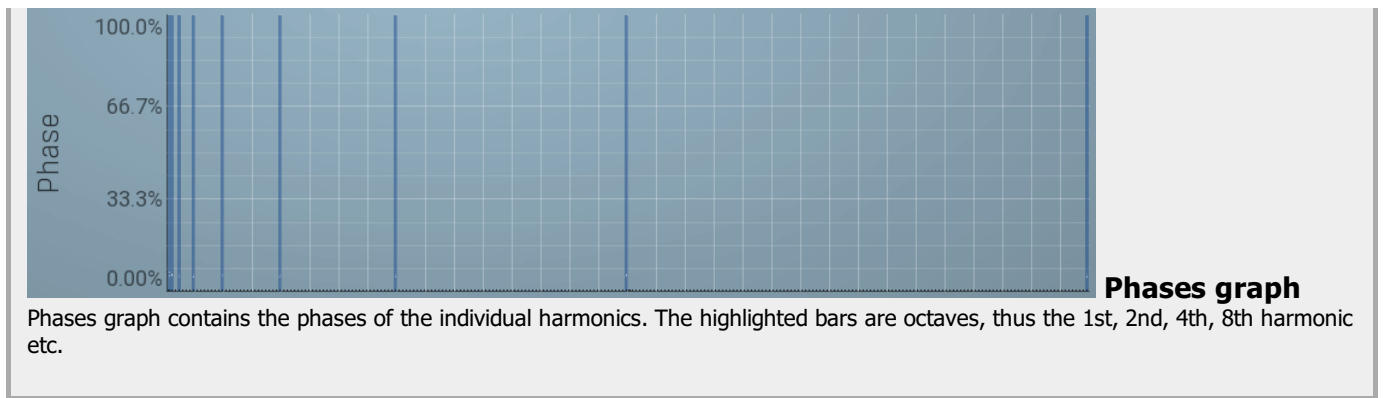
Randomize button selects random parameters for the harmonics generator, so you can use it to get a random sound character instantly. Hold **Ctrl** to slightly modify existing generator settings instead of completely changing them.



### Magnitudes

#### graph

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



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.

-3.00 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.

# Generator panel



Generator panel controls the generator engine, including all oscillators, filters and other per-voice modules.

## 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.

## Modulation sources panel

### PER-VOICE MODULATION SOURCES

Map ? ↕

Env 1 ↗	Env 2 ↗	Env 3 ↗	Env 4 ↗	Env 5 ↗	Env 6 ↗	Env 7 ↗	Env 8 ↗
Attack 1 ↗	Attack 2 ↗	Attack 3 ↗	Attack 4 ↗	Attack 5 ↗	Attack 6 ↗	Attack 7 ↗	Attack 8 ↗
LFO 1 ↗	LFO 2 ↗	LFO 3 ↗	LFO 4 ↗	LFO 5 ↗	LFO 6 ↗	LFO 7 ↗	LFO 8 ↗
Random 1 ↗	Random 2 ↗	Random 3 ↗	Random 4 ↗	Key scale 1 ↗	Key scale 2 ↗	Velocity scale 1 ↗	Velocity scale 2 ↗

Modulation sources panel contains 8 envelopes, 8 LFOs and 4 random sources, that can be used to modulate many of the parameters of the modules in the Generator. Clicking any of the buttons will open the editor for that modulation source. And dragging an arrow pointer directly onto a parameter's **Modulation** button directly assigns that source to that target parameter. Once mapped the modulation can be edited by clicking the target's parameter's **Modulation** button.

Map

### Map

Map button displays all current mappings of per-voice modulation.

Env 1 ↗

### Envelope

Envelope modulation source provides a versatile attack/decay/sustain/release envelope. Unlike traditional ADSR envelopes, this also contains features such as delay, hold, tremolo, custom shapes... Envelopes are used to create some sort of movement in the sound.



### Drag

Drag button lets you quickly associate this modulation source to a parameter by drag & dropping it onto the parameter's modulation button.

Attack 1 ↗

### Attack

Attack modulation source provides a simple freeform envelope. While you can do the same thing using envelopes in custom mode, this one is usually easier to use and less resource demanding.



### Drag

Drag button lets you quickly associate this modulation source to a parameter by drag & dropping it onto the parameter's modulation button.

LFO 1 ↗

### LFO

LFO is a versatile low-frequency oscillator. Despite it is called LFO, it can actually range to extremely high frequencies. Using very

low frequencies combined with some complex oscillator shapes these can be used to produce some sort of repeating semirandom movement as well.

### **Drag**

Drag button lets you quickly associate this modulation source to a parameter by drag & dropping it onto the parameter's modulation button.

Random 1 

### **Random**

Random source provides a per-voice random signal generator, which is useful to bring unpredictable movement to the sound. If used mildly it can be used to mimic analogue instruments.

### **Drag**

Drag button lets you quickly associate this modulation source to a parameter by drag & dropping it onto the parameter's modulation button.

Key scale 1 


### **Key scale**

Key scale lets you respond to which key is being pressed. By default lowest of the 128 MIDI keys produces value of 0%, the highest produces 100%, but you can adjust the response freely using the transformation curve.

*You may, for example, want to increase a filter frequency for higher keys. While the filters can generally follow the keys directly, you may need some sort of limited/specialized response, where the filter frequency is not constant nor following the keys 1:1.*

### **Drag**

Drag button lets you quickly associate this modulation source to a parameter by drag & dropping it onto the parameter's modulation button.

Velocity scale 1 

### **Velocity scale**

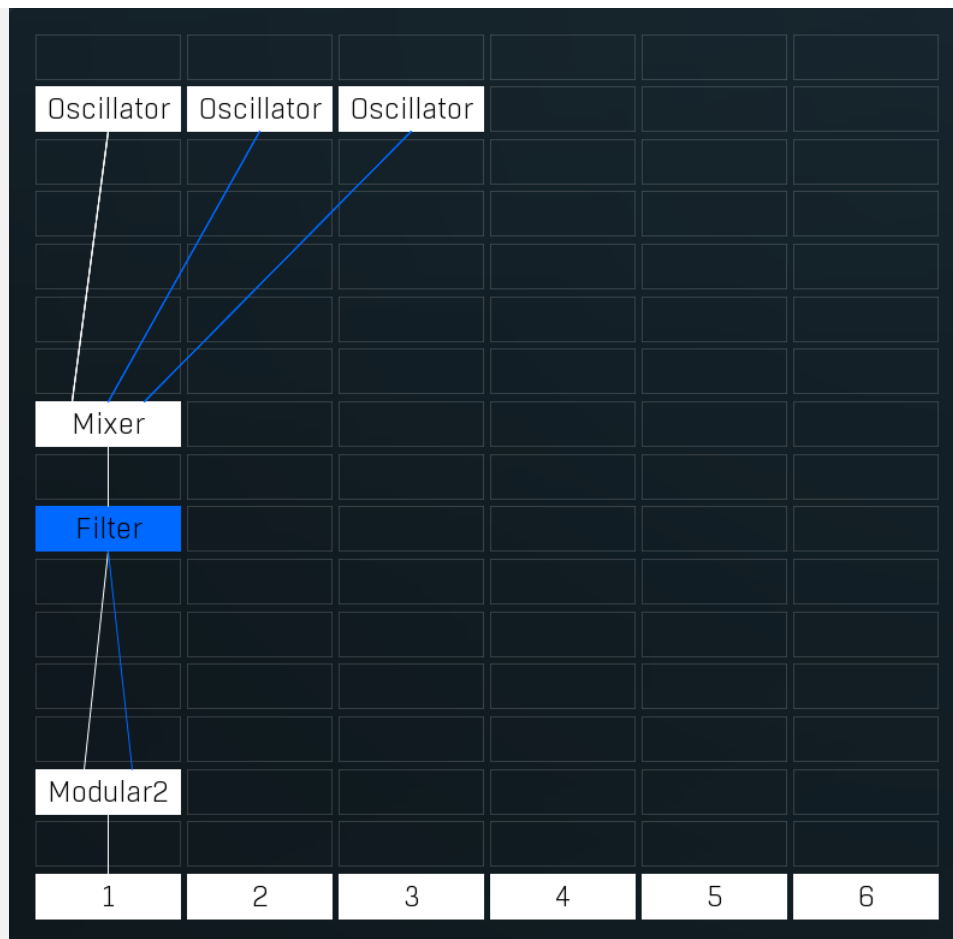
Velocity scale lets you respond to the velocity of the key being pressed. Velocity is generally defined as 'how quickly the key has been pressed'. By default minimum velocity produces 0% and maximum velocity produces 100%, but you can adjust the response freely using the transformation curve.

*You may, for example, want to control a filter frequency depending on the velocity.*

### **Drag**

Drag button lets you quickly associate this modulation source to a parameter by drag & dropping it onto the parameter's modulation button.

## 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.

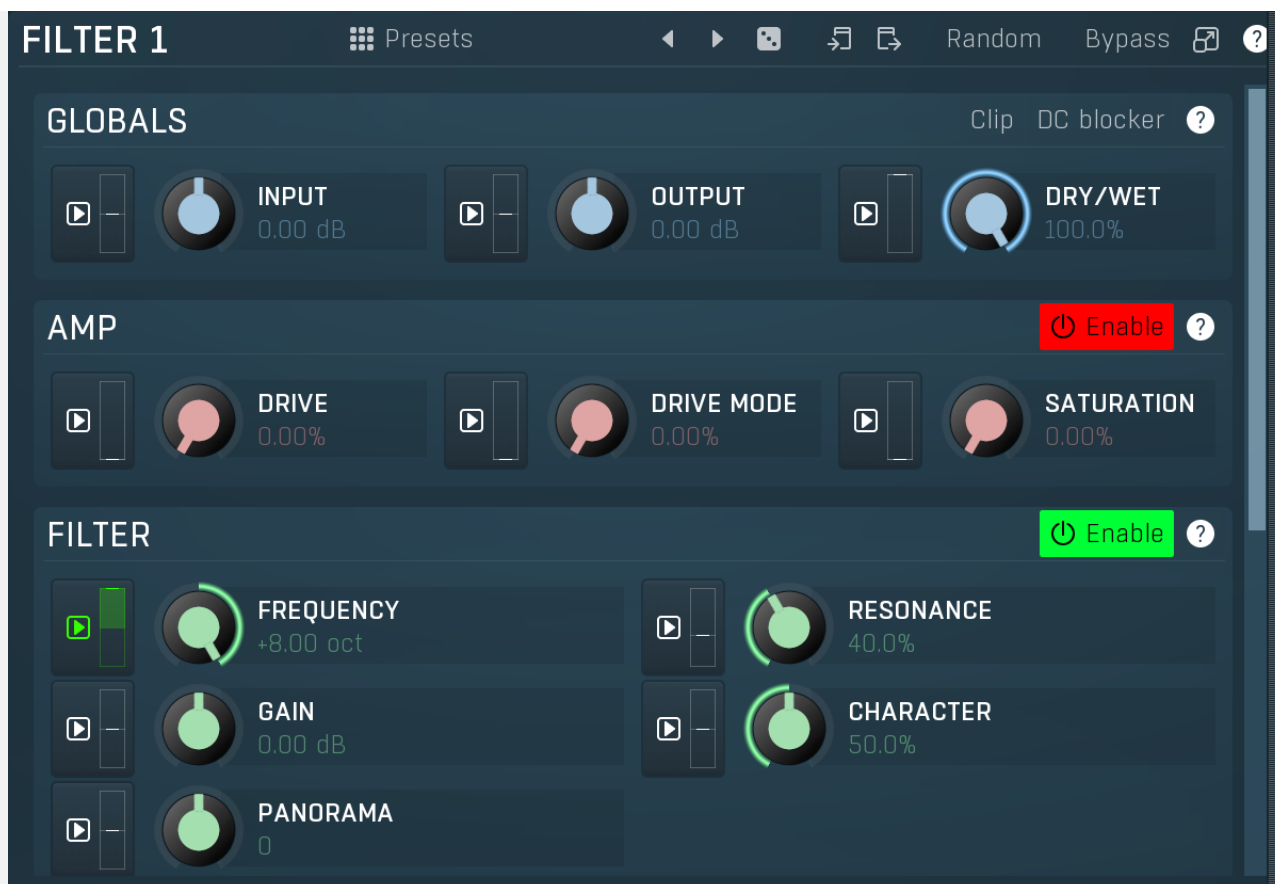
Hold **Shift** and click on the plugin to show its editor in a pop-up window.



**Collapse**

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

## Filter 1



## Presets

### 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.

## Random

### 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.



## 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.

## Globals panel



Globals panel contains settings for the overall behaviour.

## Clip

### 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.

## DC blocker

### DC blocker

DC blocker activates the filter 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.



INPUT  
0.00 dB

### Input gain

Input gain defines the input gain that is applied before the amp and the filter. This could be useful for controlling the input level, which, when too high, may cause problems especially with the amp. When you feed a 0dBFS saw wave into the amp for example, there's not much to add into the signal to enrich it, which is what an amp does, so the output signal becomes only some dirty mess and the amp may even start destroying the fundamental. This is the moment when you should probably lower the input gain to get the most out of the amp.

Range: -24.00 dB to +24.00 dB, default 0.00 dB



OUTPUT  
0.00 dB

### 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: -24.00 dB to +24.00 dB, default 0.00 dB



DRY/WET

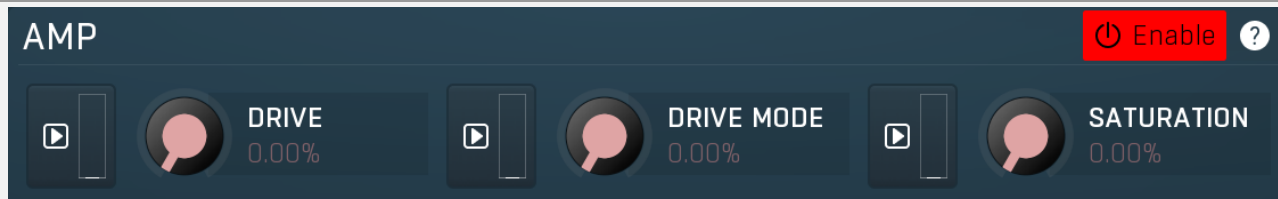
100.0%

### 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%

## Amp panel



Amp panel contains settings of the input/output amp.



DRIVE

0.00%

### 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%



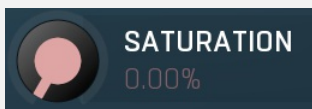
DRIVE MODE

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

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%

## Filter panel



Filter panel contains settings of the main filter processor.



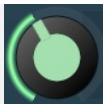
FREQUENCY

+8.00 oct

### Frequency

Frequency controls the filter minimum frequency.

Range: -8.00 oct to +8.00 oct, default 0.00 oct



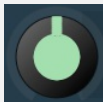
## RESONANCE

40.0%

### 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

0.00 dB

### 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

50.0%

### 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%



## PANORAMA

0

### 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: -4.00 oct to +4.00 oct, default 0

## 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).

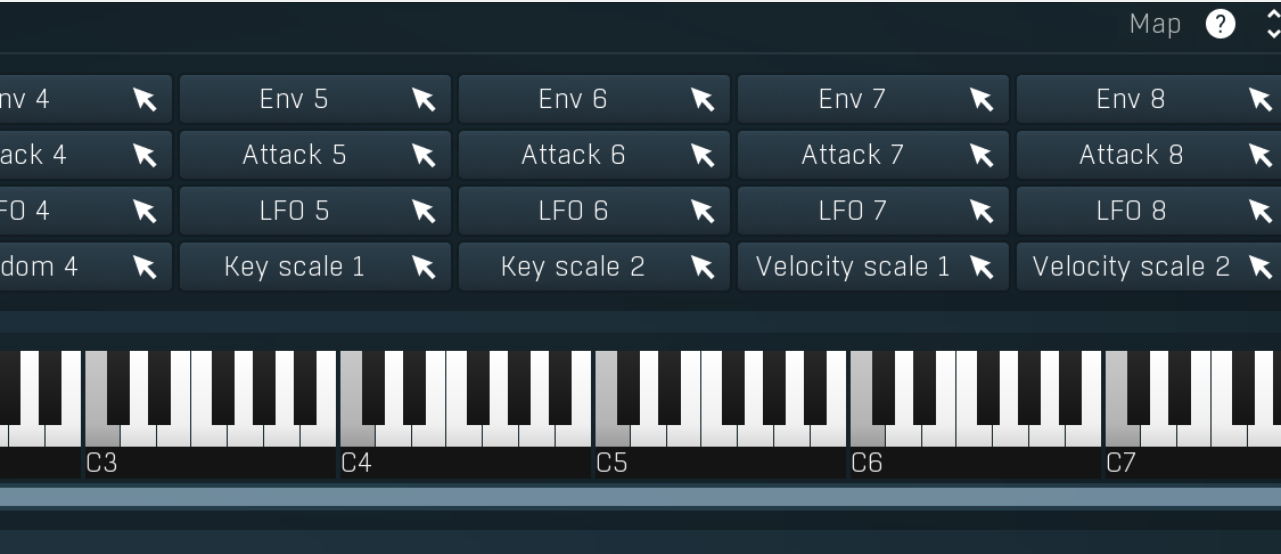
**Comb and diffuser** filters are complex comb filtering processors with pretty wild and fat responses. These range from simple comb filtering to complex almost ambient responses. Each filter uses a different kernel, so it shall be selected by trial-and-error approach. The **character** parameter controls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

Pitch mode

Pitch mode controls the way the current pitch is determined.  
**Normal** mode follows the current pitch of each voice, including glide.  
**Constant** mode ignores the note pitch and behaves as if an A4 (440Hz) note is received each time.  
**Ignore glide** mode follows the current pitch of each voice, but ignores glide.



Analyzer view

Analyzer view contains the different forms of analysis that you can display at once. This mainly consists of a powerful spectrum analyzer and sonogram.

4

Analyzer

Analyzer button enables or disables the spectrum analyzer, which shows the levels of individual frequencies. In most practical cases it is more convenient to use the sonogram, which shows the frequencies in time, but provides a lower level resolution as the levels are differentiated by color. The spectrum analyzer also provides a micro-sonogram (shown in the bottom of the panel) which uses the same color-based view as the sonogram.

Env

Fill

Fill button enables or disables the full-sized analyzer micro-sonogram. This means that the micro-sonogram at the bottom of the equalizer graph will fill the whole analyzer view. Color differentiation is often easier to understand than the classical spectrum analyzer, so this might help you better understand the spectrum of your audio material.

An alternative is to use the spectrum sonogram.

5

Analyzer Rainbow Colors

Analyzer Rainbow Colors lets you see the analyzed sound spectrum in beautiful colors, following the same style as visible light. It ranges from infra-red colors for the lowest frequencies to ultra-violet colors for the highest frequencies in the analyzed audio. If rainbow colors are disabled, the analyzer and graph will be single-colored, following the setup from Settings/Graphs.

6

Sonogram

Sonogram button enables or disables the spectrum sonogram, which shows levels of individual frequencies in time. Levels are differentiated by color, so the accuracy is not as good as when using the spectrum analyzer. However, the time axis improves the visual orientation in the spectrum for typical audio signals. In contrast, the spectrum analyzer is more of a scientific tool.

7

Settings

Settings button shows the settings of the spectrum analyzer and the spectrum sonogram.

Analyzer settings



 Presets

## 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.

**MAIN SETTINGS**

ADVANCED

GRAPHS

SONOGRAM

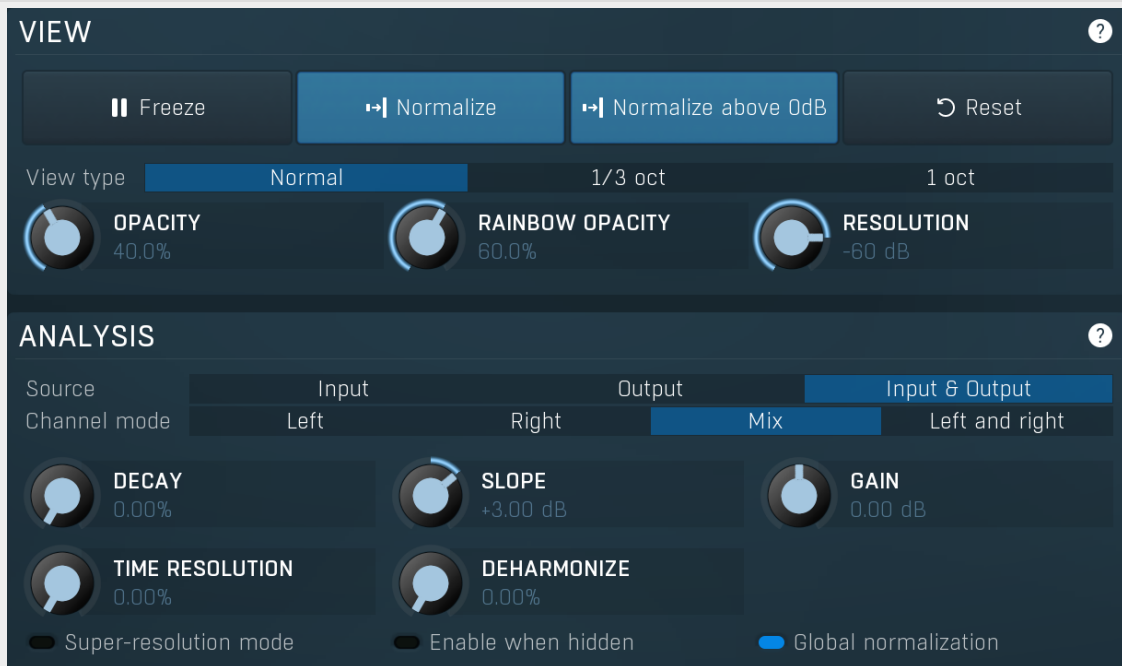
PREFILTERING

**Tab**

**selector**

Tab selector switches between subsections.

## Main settings panel



Main settings panel contains the most useful settings controlling the analyzer behaviour and view.

## View



Freeze

### Freeze

Freeze button stops processing temporarily.

Normalize

### Normalize

Normalize button enables or disables the visual normalization, which makes the loudest frequency be displayed at the top of the analyzer area (0dB); it does not normalise the sound. This is very useful for comparing frequency levels, however it does hide the actual level.

When comparing 2 spectrums you are usually interested mainly in the frequency level differences. In most cases both audio materials will have different overall levels, which would mean that one of the graphs would be "lower" than the other, making the comparison quite difficult. Normalize fixes this and makes the most prominent frequencies of the spectrum reach the top of the analyzer area (or have the most highlighted color in case of sonogram).

Reset

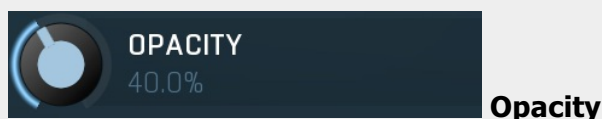
### Reset

Reset button resets the analyzer state. This is particularly useful when analyzing infinite average and maximum values.

View type **Normal** 1/3 oct 1 oct **View**

### type

View type controls the way the spectrum is displayed. By default a smooth curve is presented. This view provides the best resolution and detail, but other modes (1/3 octave, 1 octave) may be easier to read.



Opacity

Opacity controls the opacity of all analyzer graphs.



### Rainbow opacity

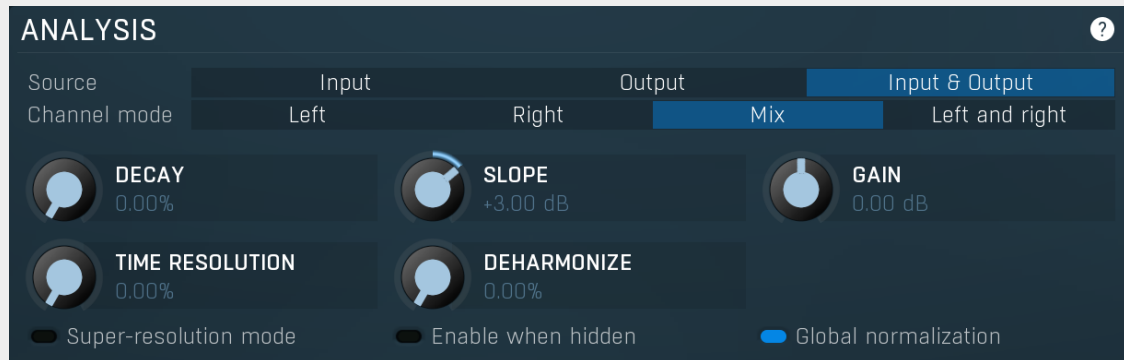
Rainbow opacity controls the opacity of the rainbow graph, if enabled.



### Resolution

Resolution defines the vertical range on the display. The human auditory system has a resolution of about 90dB and the relevant range is usually less than 60dB. However you may want to use a higher resolution to check for technical problems - aliasing, distortion etc.

## Analysis



Source mode defines which audio stages are to be analyzed. By default both input & output are selected and analyzed. However you may want to analyze only the input, or the output (or the external side-chain, where available, on its own or with the input or output).

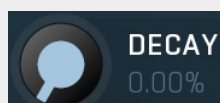
Channel mode defines which channels are to be analyzed. By default all channels are merged into a mono sum (Mix mode), which is then analyzed. However you may want to analyze separate channels or display both the left and right channels separately. Please note that if two channels (for example: input & output, or input & side-chain) are displayed at the same time then mix mode is used instead of left & right mode. Similarly, when the plug-in is in Surround mode then Mix mode is used.

### Channel mode

Channel mode defines which channels are to be analyzed. By default all channels are merged into a mono sum (Mix mode), which is then analyzed. However you may want to analyze separate channels or display both the left and right channels separately. Please note that if two channels (for example: input & output, or input & side-chain) are displayed at the same time then mix mode is used instead of left & right mode. Similarly, when the plug-in is in Surround mode then Mix mode is used.

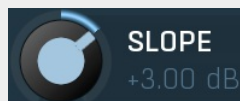
Also please note that when the plug-in is in one of the Mid / Side modes of operation, then you should read 'Left' as 'Mid' and 'Right' as 'Side'.

Different analyser combinations can, of course, be saved as different named presets.



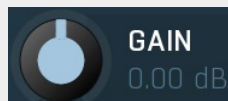
### Decay

Decay controls the speed at which the magnitudes return to the minimum value (silence). It is an alternative to averaging, which affects the speed that the frequencies both gain and lose their magnitudes. With a decay of 0% the magnitude goes to the minimum immediately. With 100% it stays the same forever, so it makes it display the maximum.



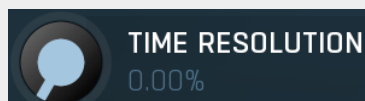
### Slope

Slope makes the analyser increase the magnitude of higher frequencies, since they are typically lower in energy. 3dB per octave is a typical value, which makes pink noise horizontal as pink noise contains equal energy in each octave. Therefore if you set slope to 3dB, the response would be the same for the FFT and 1/3 octave graphs.



### Gain


Gain makes all frequencies change magnitude by the specified amount. This has no meaning when normalization is enabled.



### Time resolution



Time resolution improves the time resolution, but lowers the spectral resolution. This is typically useful for more scientific analyses, where the signal is moving quickly and you need to follow its movements quickly. This is often advantageous for sonograms with very high FFT sizes.

 **DEHARMONIZE**  
0.00%

### Deharmonize

Deharmonize tries to remove harmonics in the content and leave only fundamentals. This may help you find the dominant frequencies in the signal.

☐ Super-resolution mode

### Super-resolution mode

Super-resolution mode activates a special processing algorithm, which provides high resolution even in the low frequency spectrum. Using standard FFT algorithms you can increase the FFT size to get better bass resolution, but this also slows down the response. Super-resolution mode keeps the quick response in high frequencies as they are naturally quicker, but also highly enhances the bass spectrum resolution. It requires additional CPU power.

☐ Enable when hidden

### Enable when hidden

Enable when hidden causes the analysis engine to continue processing the signal even when the GUI is hidden. Otherwise the sonogram is stopped, therefore will not be immediately available when the GUI is shown again.


☒ Global normalization


### Global normalization

Global normalization makes the **normalization** work based on the maximum of all graphs visible at the time. This means that the levels between the graphs will stay the same, but the maximum level will be 0dB. This is useful for comparing relative levels. If you disable this, all graphs will be normalized separately and will touch 0dB unless they are silent; and this is useful for comparing spectra.

## Advanced panel

**PEAK DETECTION** ?

 **PEAK DETECTION**  
0.00%

 **PEAK THRESHOLD**  
-40.00 dB

**SCIENTIFIC SETTINGS** ?

Overlap

Window type

☐ Analytical smoothing

FFT size

☒ Logarithmic averaging

**APPEARANCE**


Axis color


Grid color

Advanced panel contains more advanced settings controlling the scientific parameters of the audio analysis.

## Peak detection

**PEAK DETECTION** ?

 **PEAK DETECTION**  
0.00%

 **PEAK THRESHOLD**  
-40.00 dB

 **PEAK DETECTION**  
0.00%

### Peak detection

Peak detection tries to the remove skirts of separate sinusoids letting you view the frequencies contained in your audio material. This may be handy when performing more scientific analyses.

 **PEAK THRESHOLD**  
-40.00 dB

### Peak threshold

Peak threshold defines the level below the maximum which is used for peak detection. You can use this to control which peaks get through and to get rid of small insignificant ones.

## Scientific settings

## SCIENTIFIC SETTINGS

Overlap	4x	◀ ▶	FFT size	4096	◀ ▶
Window type	Hann	◀ ▶			
<input type="checkbox"/> Analytical smoothing			<input checked="" type="checkbox"/> Logarithmic averaging		

### Overlap 4x Overlapping

Overlapping makes the analyser perform multiple FFT processing on the same data which results in better precision at the cost of higher CPU impact. With higher overlapping the response also speeds up.

### FFT size 4096 FFT size

FFT size defines FFT processing block size. It basically controls the resolution. However for higher resolution in bass content it is recommended to use super-resolution mode instead as it keeps the quick response in higher frequencies.

### Window type Hann Window type

Window type defines the type of window used to pre-process the source samples. This has several consequences for the frequency response, but it is a little scientific parameter. If you do not have specific requirements you can just leave this set to its default.

### ☐ Analytical smoothing Analytical smoothing

Analytical smoothing switch activates a more complicated smoothing algorithm, which provides more accurate results, however it may require much more CPU power. Unlike normal smoothing this method doesn't change the proportions of frequencies with higher magnitudes. It is useful mostly for technical analysis and for most musical signals it is often better to use the default smoothing method.

### ☒ Logarithmic averaging Logarithmic averaging

Logarithmic averaging switch activates averaging in logarithmic mode, hence decibels. If you disable it, linear averaging will be used.

## Graphs panel

<b>AVERAGE</b>		Copy analysis		<input checked="" type="checkbox"/> Enable	?	⬆ ⬇ ⬆
<input type="checkbox"/> Peaks	Line opacity	100.0%				
<input type="checkbox"/> Micro-sonogram	Line width	1				
<input type="checkbox"/> Sonogram fill	Fill opacity	100.0%				
<b>AVERAGE (INFINITE)</b>		Copy analysis		<input type="checkbox"/> Enable	?	⬆ ⬇ ⬆
<input type="checkbox"/> Peaks	Line opacity	100.0%				
<input type="checkbox"/> Micro-sonogram	Line width	1				
<input type="checkbox"/> Sonogram fill	Fill opacity	100.0%				
<b>MAXIMUM</b>		Copy analysis		<input type="checkbox"/> Enable	?	⬆ ⬇ ⬆
<input type="checkbox"/> Peaks	Line opacity	100.0%				
<input type="checkbox"/> Micro-sonogram	Line width	1				
<input type="checkbox"/> Sonogram fill	Fill opacity	100.0%				
<b>MAXIMUM (INFINITE)</b>		Copy analysis		<input type="checkbox"/> Enable	?	⬆ ⬇ ⬆
<input type="checkbox"/> Peaks	Line opacity	100.0%				
<input type="checkbox"/> Micro-sonogram	Line width	1				
<input type="checkbox"/> Sonogram fill	Fill opacity	100.0%				
<b>MAXIMUM - AVERAGE (INFINITE)</b>		Copy analysis		<input type="checkbox"/> Enable	?	⬆ ⬇ ⬆
<input type="checkbox"/> Peaks	Line opacity	100.0%				
<input type="checkbox"/> Micro-sonogram	Line width	1				
<input type="checkbox"/> Sonogram fill	Fill opacity	100.0%				
<b>COMPARISON</b>		Paste analysis		<input type="checkbox"/> Enable	?	⬆ ⬇ ⬆
<input type="checkbox"/> Peaks	Line opacity	100.0%				
<input type="checkbox"/> Micro-sonogram	Line width	2				
<input type="checkbox"/> Sonogram fill	Fill opacity	30.0%				

Graphs panel contains visual settings for the different graphs that you can show in the analyzer.

Average

AVERAGE

Copy analysis

Enable

?

Peaks

Micro-sonogram

Sonogram fill

Line opacity

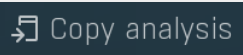
Line width

Fill opacity

100.0%

1

100.0%



Copy analysis

Copy analysis button copies the current state of the analysis into the system clipboard so that you can paste it into another analyzer for comparison. Hold **ctrl** to export the analysis into a CSV file.



Analyzer Fill and Line Color - All Channels

Analyzer Fill and Line Color - All Channels defines the fill and line colors of the analyzer graph when Rainbow analyzer colors are disabled. This setting is used to set colors in all channel modes - see channel mode help in the Main Settings tab.



Analyzer Fill and Line Color - Left Channel

Analyzer Fill and Line Color - Left Channel defines the fill and line colors of the analyzer graph when Rainbow analyzer colors are disabled. This setting is used to set colors in the left channel mode only - see channel mode help in the Main Settings tab.



Peaks

Peaks enables detection of frequencies with the highest magnitudes. Frequencies which are at most 20dB lower than the maximum are displayed, and there may be at most 8 of them. Please note that this feature requires additional CPU power.

Line opacity

100.0%

Line opacity

Line opacity controls the opacity of the graph outline.



Micro-sonogram

Micro-sonogram displays a small single-state sonogram at the bottom of the graph. This may help you compare relevant frequencies, because it is usually easier to compare colors than graph values.

Line width

1

Line width

Line width controls the width of the graph outline.



Fill

Fill makes the sonogram (enabled by **Show sonogram**) fill the whole area.

Fill opacity

100.0%

Fill opacity

Fill opacity controls the opacity of the graph interior fill.

Average (infinite)

AVERAGE (INFINITE)

Copy analysis

Enable

?

Peaks

Micro-sonogram

Sonogram fill

Line opacity

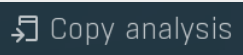
Line width

Fill opacity

100.0%

1

100.0%



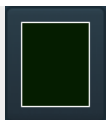
Copy analysis

Copy analysis button copies the current state of the analysis into the system clipboard so that you can paste it into another analyzer for comparison. Hold **ctrl** to export the analysis into a CSV file.



Analyzer Fill and Line Color - All Channels

Analyzer Fill and Line Color - All Channels defines the fill and line colors of the analyzer graph when Rainbow analyzer colors are disabled. This setting is used to set colors in all channel modes - see channel mode help in the Main Settings tab.

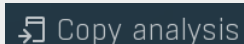


### Analyzer Fill and Line Color - Left Channel

Analyzer Fill and Line Color - Left Channel defines the fill and line colors of the analyzer graph when Rainbow analyzer colors are disabled. This setting is used to set colors in the left channel mode only - see channel mode help in the Main Settings tab.

## Maximum

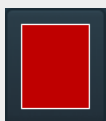
MAXIMUM		Copy analysis	Enable	?	⬆
<input type="radio"/> Peaks	Line opacity	100.0%			
<input type="radio"/> Micro-sonogram	Line width	1			
<input type="radio"/> Sonogram fill	Fill opacity	100.0%			



Copy analysis

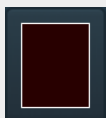
### Copy analysis

Copy analysis button copies the current state of the analysis into the system clipboard so that you can paste it into another analyzer for comparison. Hold **ctrl** to export the analysis into a CSV file.



### Analyzer Fill and Line Color - All Channels

Analyzer Fill and Line Color - All Channels defines the fill and line colors of the analyzer graph when Rainbow analyzer colors are disabled. This setting is used to set colors in all channel modes - see channel mode help in the Main Settings tab.

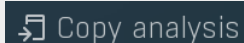


### Analyzer Fill and Line Color - Left Channel

Analyzer Fill and Line Color - Left Channel defines the fill and line colors of the analyzer graph when Rainbow analyzer colors are disabled. This setting is used to set colors in the left channel mode only - see channel mode help in the Main Settings tab.

## Maximum (infinite)

MAXIMUM (INFINITE)		Copy analysis	Enable	?	⬆
<input type="radio"/> Peaks	Line opacity	100.0%			
<input type="radio"/> Micro-sonogram	Line width	1			
<input type="radio"/> Sonogram fill	Fill opacity	100.0%			



Copy analysis

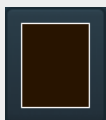
### Copy analysis

Copy analysis button copies the current state of the analysis into the system clipboard so that you can paste it into another analyzer for comparison. Hold **ctrl** to export the analysis into a CSV file.



### Analyzer Fill and Line Color - All Channels

Analyzer Fill and Line Color - All Channels defines the fill and line colors of the analyzer graph when Rainbow analyzer colors are disabled. This setting is used to set colors in all channel modes - see channel mode help in the Main Settings tab.

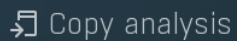


### Analyzer Fill and Line Color - Left Channel

Analyzer Fill and Line Color - Left Channel defines the fill and line colors of the analyzer graph when Rainbow analyzer colors are disabled. This setting is used to set colors in the left channel mode only - see channel mode help in the Main Settings tab.

## Maximum - Average (infinite)

MAXIMUM - AVERAGE (INFINITE)		Copy analysis	Enable	?	⬆
<input type="radio"/> Peaks	Line opacity	100.0%			
<input type="radio"/> Micro-sonogram	Line width	1			
<input type="radio"/> Sonogram fill	Fill opacity	100.0%			



Copy analysis

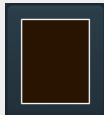
### Copy analysis

Copy analysis button copies the current state of the analysis into the system clipboard so that you can paste it into another analyzer for comparison. Hold **ctrl** to export the analysis into a CSV file.



### Analyzer Fill and Line Color - All Channels

Analyzer Fill and Line Color - All Channels defines the fill and line colors of the analyzer graph when Rainbow analyzer colors are disabled. This setting is used to set colors in all channel modes - see channel mode help in the Main Settings tab.



### Analyzer Fill and Line Color - Left Channel

Analyzer Fill and Line Color - Left Channel defines the fill and line colors of the analyzer graph when Rainbow analyzer colors are disabled. This setting is used to set colors in the left channel mode only - see channel mode help in the Main Settings tab.

## Comparison

COMPARISON		Paste analysis	Enable	?	↕
<input checked="" type="checkbox"/> Peaks	Line opacity	<div><div></div></div>	100.0%		
<input checked="" type="checkbox"/> Micro-sonogram	Line width	<div><div></div></div>	2		
<input checked="" type="checkbox"/> Sonogram fill	Fill opacity	<div><div></div></div>	30.0%		



Paste analysis

### Paste analysis

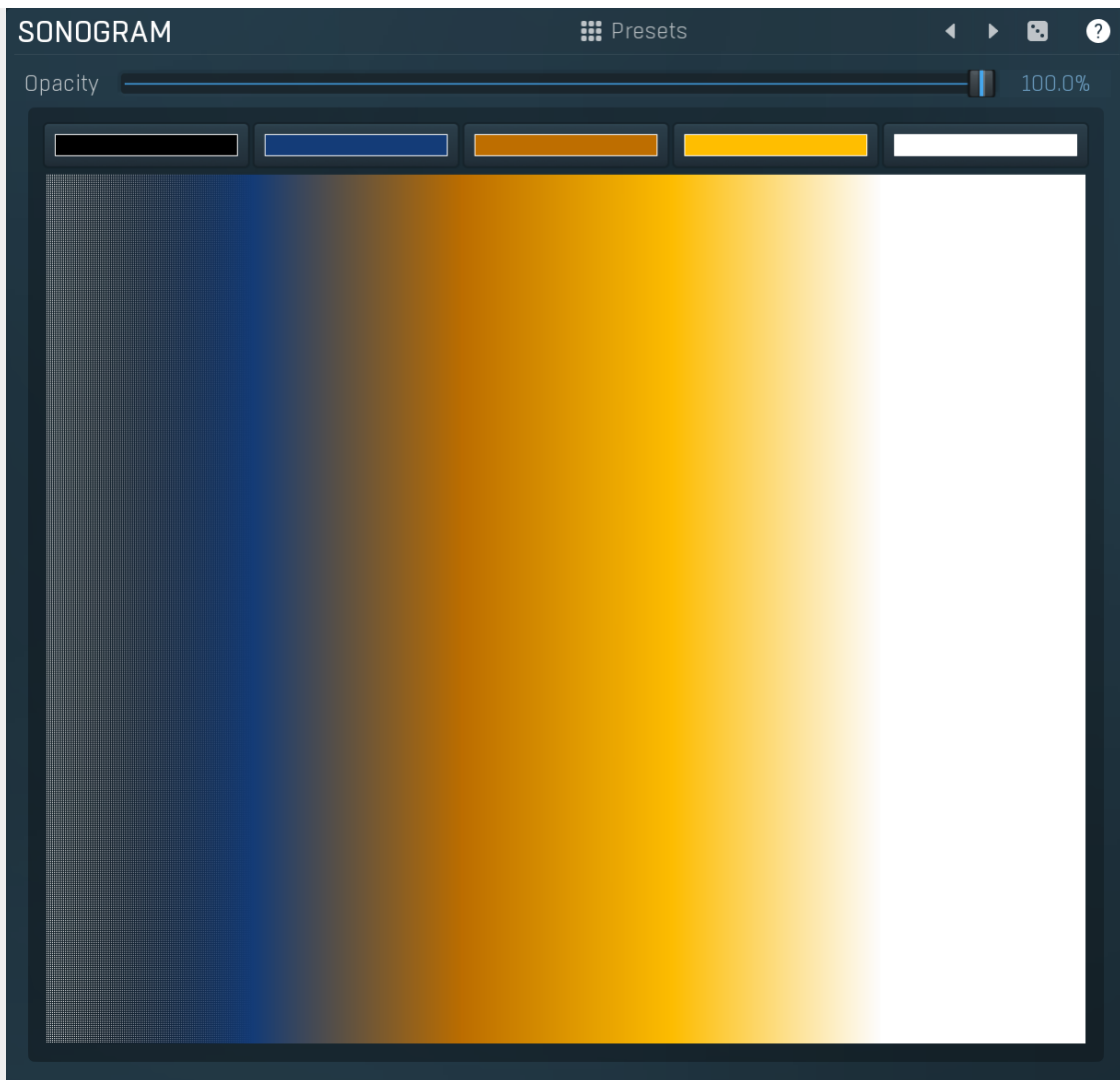
Paste analysis button pastes an analysis from the system clipboard and displays it as a comparison. This way you can compare your analysis to any other analysis from MeldaProduction plugins.



### Analyzer Fill and Line Color - Left Channel

Analyzer Fill and Line Color - Left Channel defines the fill and line colors of the analyzer graph when Rainbow analyzer colors are disabled. This setting is used to set colors in the left channel mode only - see channel mode help in the Main Settings tab.

## Sonogram panel



Sonogram panel contains visual settings of the sonogram, mainly the sonogram colors. A sonogram uses a set of colors. When the particular frequency's level is at the minimum, the first color is used. When it is at the maximum, the last color is used. Otherwise it interpolates the colors in-between.



### 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.



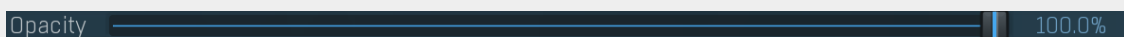
### Right arrow

Right arrow button loads the next preset.



### Randomize

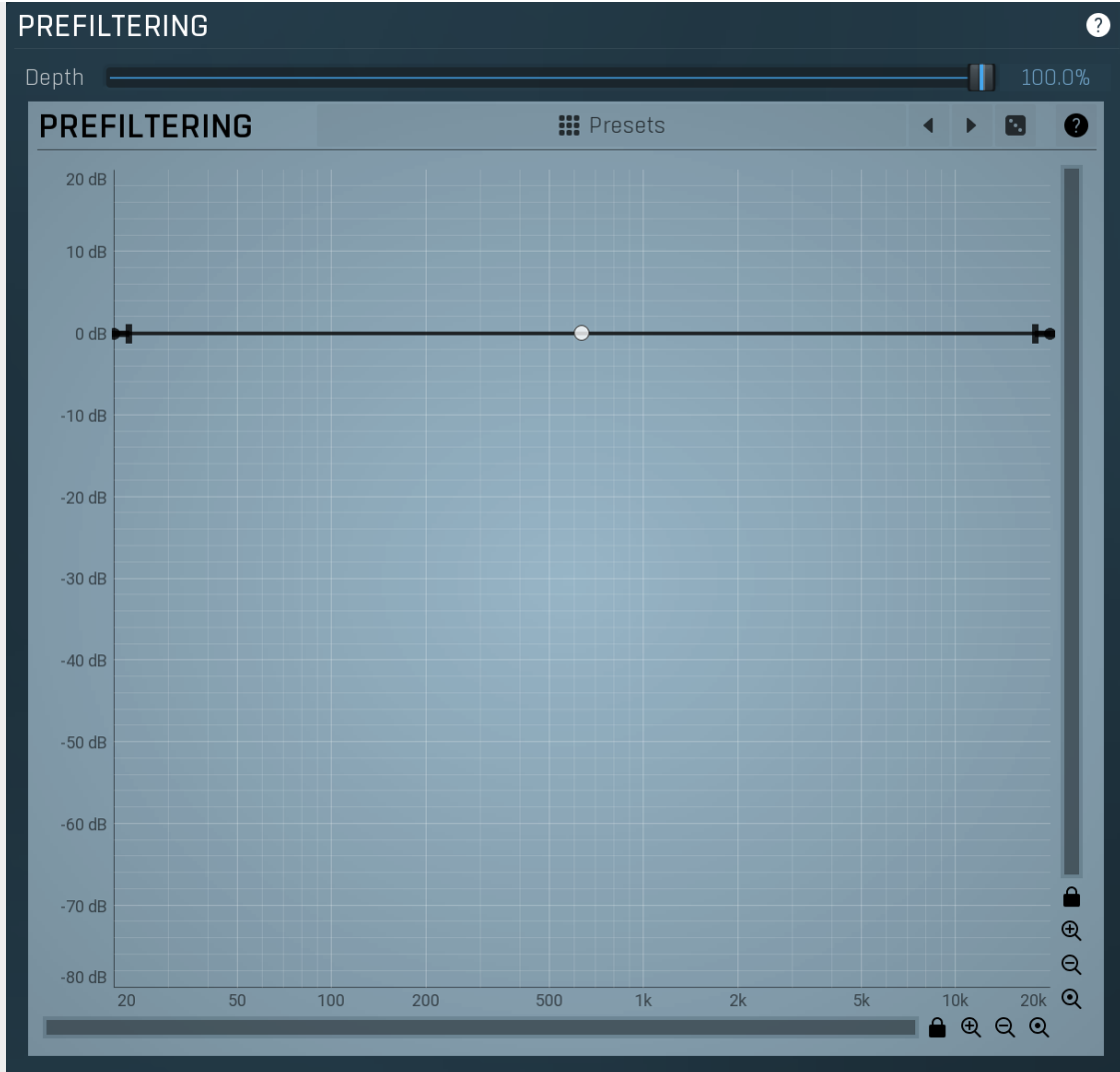
Randomize button loads a random preset.



### Opacity

Opacity controls the opacity of the sonogram.

## Prefiltering panel

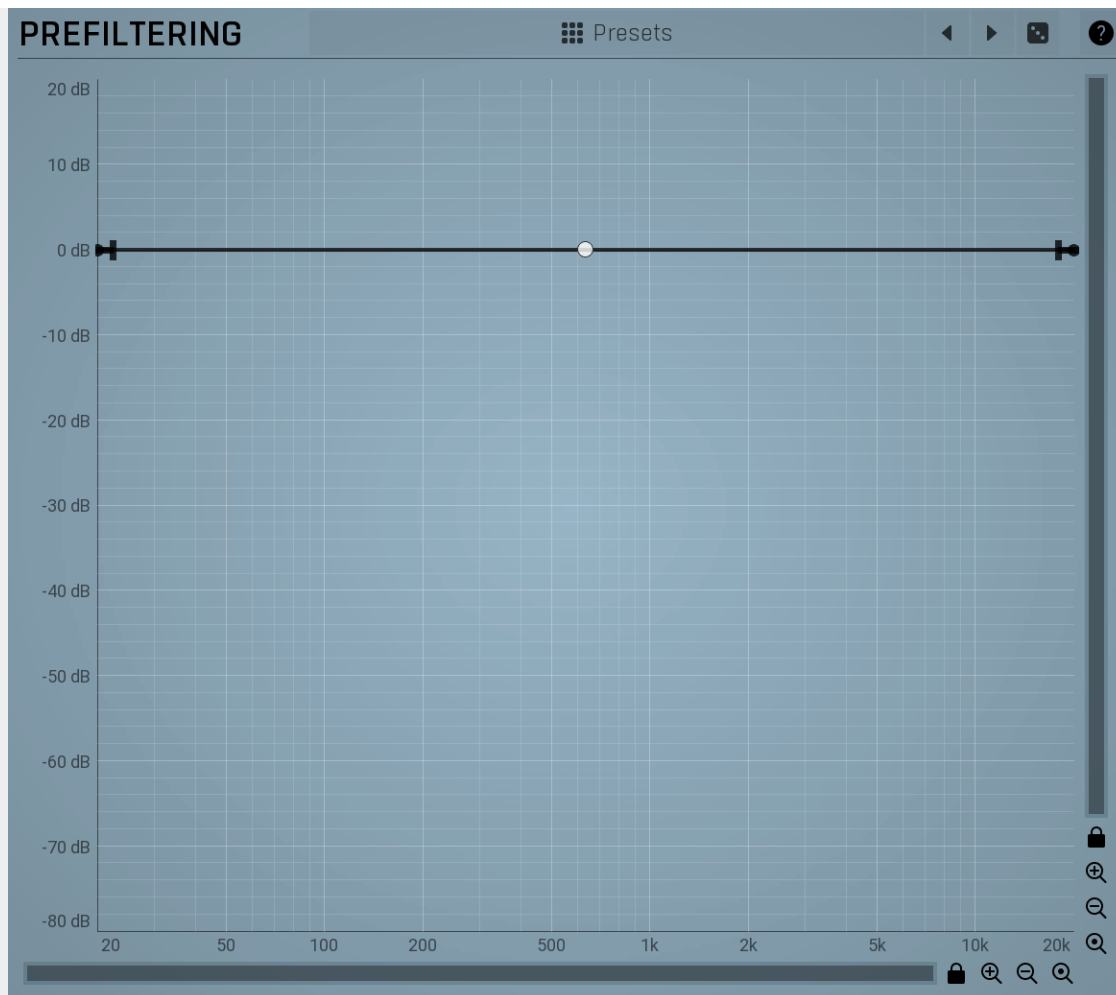


Prefiltering panel provides the optional prefiltering, which means that level of each frequency is either increased or decreased before analysis. Normally the analyzer shows scientific levels of each frequency. However you can for example use the predefined loudness curves, which makes the analyzer show how the human auditory system responds to the frequencies, so it in fact provides more accurate analysis taking into account the fact that human hearing is more complicated than the mathematical model.

**Depth** 100.0% **Depth**

Depth controls the amount of prefiltering. 100% makes the analyzer follow the prefiltering graph precisely, 0% essentially disables this feature.






## Prefiltering


### 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.

nv 7

## Pause

Pause button stops the analyzer temporarily.



## Normalize

Normalize button enables or disables the visual normalization, which makes the loudest frequency be displayed at the top of the analyzer area (0dB); it does not normalise the sound. This is very useful for comparing frequency levels, however it does hide the actual level.

When comparing 2 spectrums you are usually interested mainly in the frequency level differences. In most cases both audio materials will have different overall levels, which would mean that one of the graphs would be "lower" than the other, making the comparison quite difficult. Normalize fixes this and makes the most prominent frequencies of the spectrum reach the top of the analyzer area (or have the most highlighted color in case of sonogram).



## Reset

Reset button resets analyzer graphs. This is particularly useful when analyzing infinite average and maximum values.



## Copy analysis

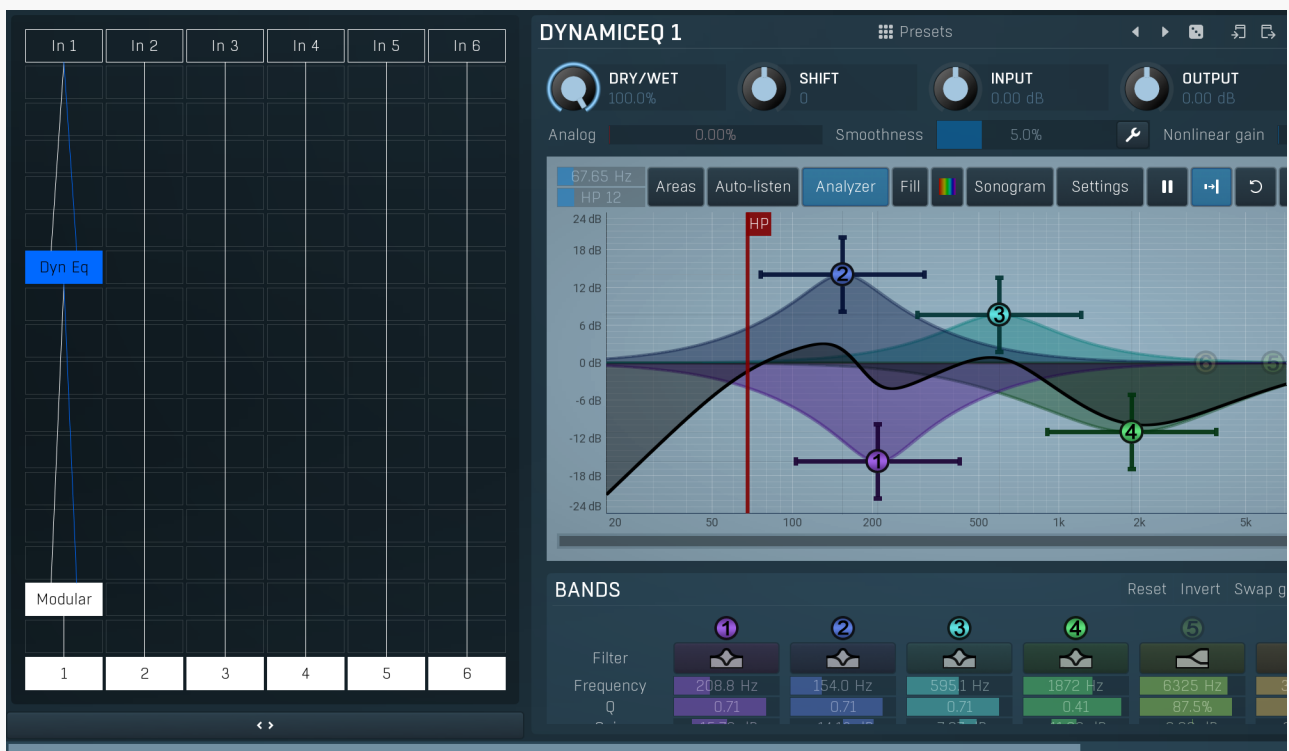
Copy analysis button copies the current state of the analysis into the system clipboard so that you can paste it into another analyzer for comparison. Hold **ctrl** to export the analysis into a CSV file.



## Paste

Paste button pastes the analysis from the system clipboard and displays it as the comparison in the graph.

# 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 pop-up 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

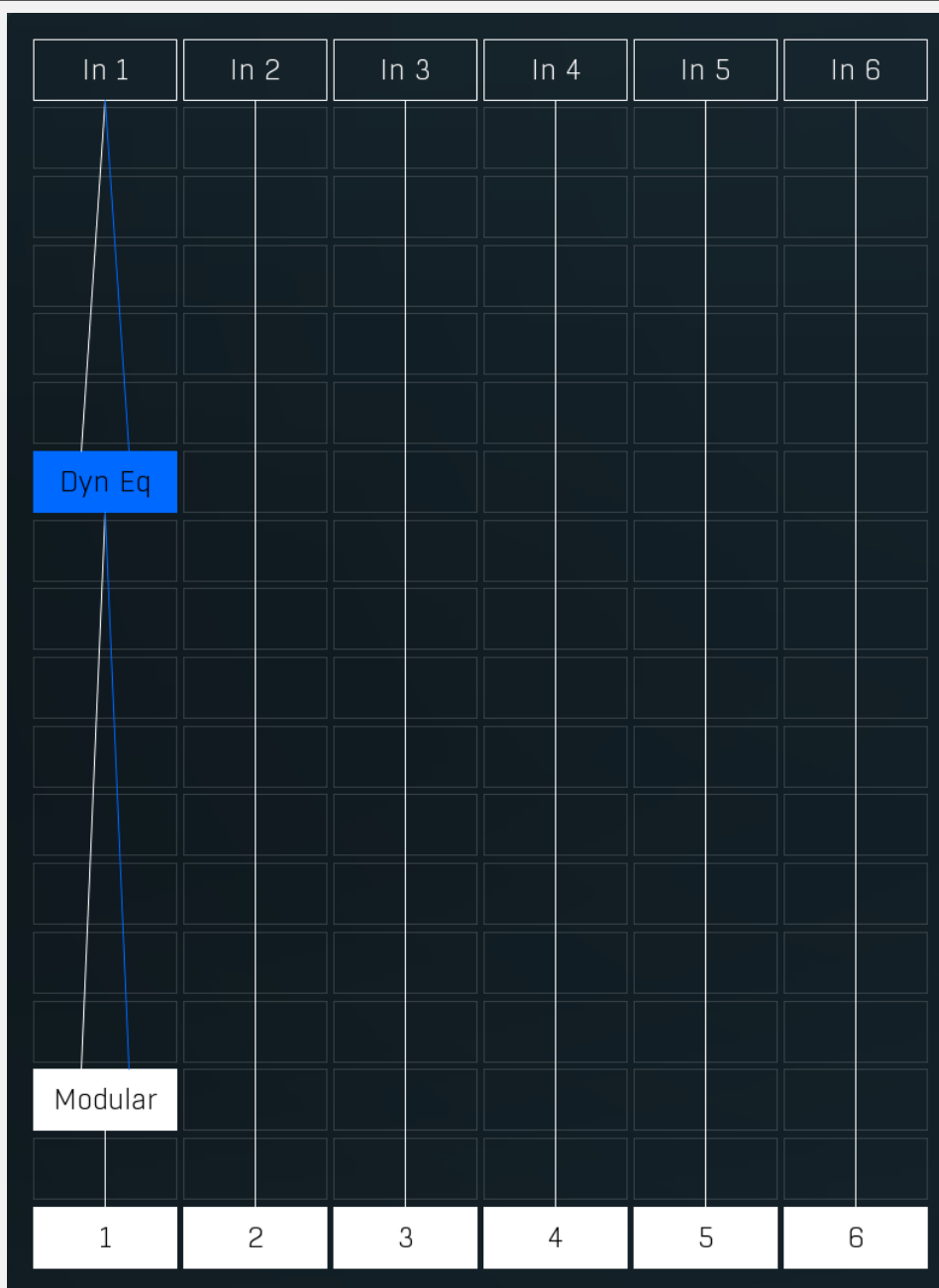
unwanted effect. Our plugins employ a smart randomization engine that learns which settings are suitable for randomization (using the existing presets) and so is much more likely to create successful changes.

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.

## DynamicEq 1



Presets

### 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.



### 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

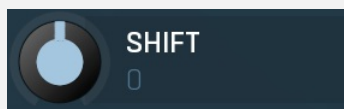
Show window button displays the plugin in a dedicated pop-up window. You can do the same thing by Shift + click on the processor item in the modular grid.



## Dry/Wet

Dry/Wet defines ratio between dry and wet signals. 100% means fully processed, 0% means no processing at all. In normal mode only peak and shelf filters are affected correctly, other filters are left at 100% unless the ratio is set to 0%, in which case the equalizer is bypassed.

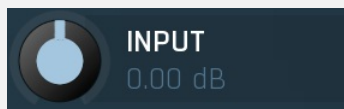
Range: 0.00% to 100.0%, default 100.0%



## Shift

Shift lets you pitch shift all bands by specified number of semitones. It doesn't change the actual band points, but changes the resulting EQ shape appropriately.

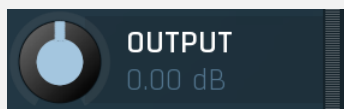
Range: -24.00 to +24.00, default 0



## Input gain

Input gain defines input gain applied before the equalization. Therefore this affects all dynamics-based processes.

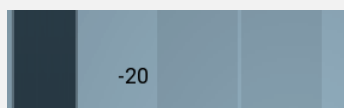
Range: -24.00 dB to +24.00 dB, default 0.00 dB



## Output gain

Output gain defines output gain applied after the equalization. Please note that the real output gain is affected by dry/wet parameter, as opposed to input gain, which is not, because it affects the resulting sound.

Range: -24.00 dB to +24.00 dB, default 0.00 dB



## Soft saturation

Soft saturation defines amount of saturation simulating analog equalizers.

Range: 0.00% to 100.0%, default 0.00%



## Analog

Analog controls the amount of internal nonlinearities in each filter, typical for analog equalizers. Note that this processing also changes the actual filter shapes, so they won't fully match the displayed graphs anymore when Analog feature is used. It can also require a solid amount of CPU power.

Range: 0.00% to 100.0%, default 0.00%



## Smoothness

Smoothness makes the analyzer smooth out the curve, so it contains less bumping up and down. It approximates the energy in each frequency and the resulting graph should be easier to understand. Also the smoothness affects the automatic equalization. Usually higher value provides more natural results, however you should verify using your ears.

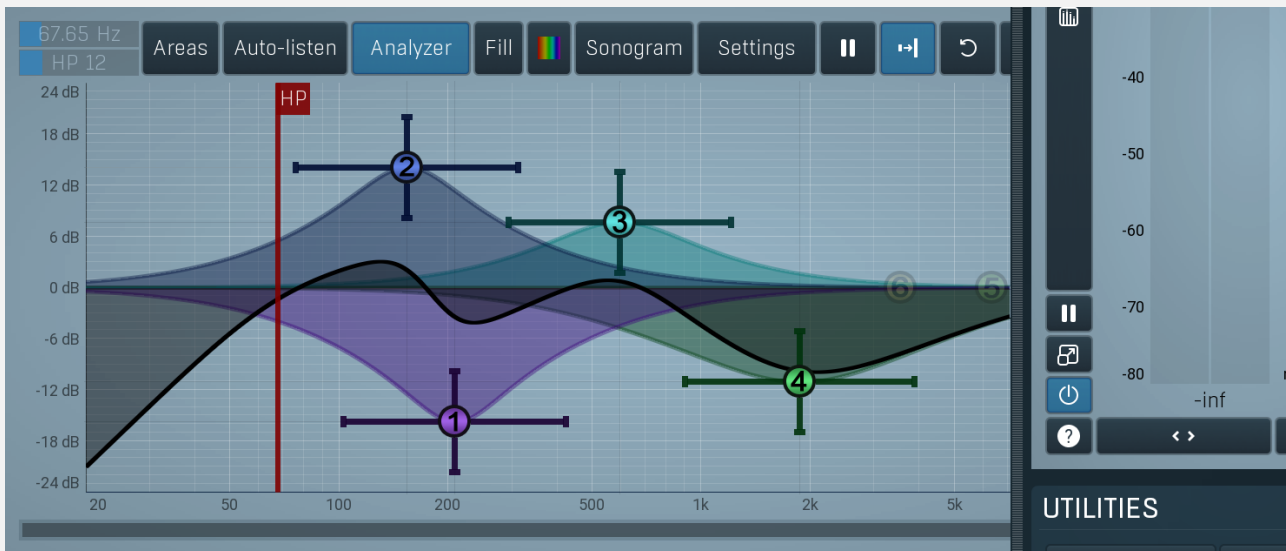
Range: 0.00% to 20.0%, default 5.0%



## Nonlinear gain

Nonlinear gain changes the way the band gain is interpreted. Normally the gain is assumed to be exactly as you set it. But in some cases, e.g. during mastering, you may focus on very small gains and that's exactly what this feature does.

Range: 0.00% to 100.0%, default 0.00%



## Equalizer shape graph

Equalizer shape graph controls and displays the frequency response. There are several bands available, each of them can be enabled/disabled, can be set to a different filter, can have different frequency, Q and other parameters.

Double-click on a band point to enable or disable a band. Drag it to change its frequency and gain. Drag the horizontal nodes to change its Q. Hold **ctrl** key for fine tuning. Click using the right mouse button on it to open a window with additional settings.

The equalizer graph also contains 2 **red vertical lines** on the right and on the left. These are the high-pass and low-pass filters, conveniently placed so that you can perform this often-used task quickly, efficiently and most of all using the highest quality filters available on the market. The high-pass filter also serves as a DC blocker. Slopes for both filters can be adjusted in the title area of the equalizer graph panel.

Typically you want to remove the low frequencies (high-pass filter via the left line) from just about any audio material except for bass and bass drum. Even if the frequencies are not there and are not shown in the sonogram or analyzer views, you may still want to apply a high-pass filter to let the equalization remove any potential low frequency rumble the track might contain. This is always a good practice to clear the resulting mix.

### Areas

#### Areas

Areas button displays settings for the visual areas, which are useful for better visual orientation in the frequency spectrum. These areas are customisable guidelines displayed in the equalizer editor and may contain different octave bands or typical drum frequencies for example. Note that these areas are always only guides, so your particular snare drum may not fit exactly in the very well with the example. In that case it is highly advantageous to use the sonogram or analyzer. Or you can edit your own areas.

### Auto-listen

#### Auto-listen

Auto-listen button enables the auto-listen feature, which temporarily changes the equalizer shape when dragging a band to let you see and hear what that particular band is actually doing. For example, when dragging a peak filter, the equalizer disables the other bands and changes this one to a band pass filter, so that you can focus on the frequencies that the peak filter is modifying.

Also, when this is enabled, you can click anywhere in the band's area (shaded) and the equalizer will let you listen to the frequencies at that position using a band-pass filter. This is great for searching for problematic frequencies for example. Vertical position controls the band-width. You can also hold **shift** to get this feature if auto-listen is not enabled.

### Analyzer

#### Analyzer

Analyzer button enables or disables the spectrum analyzer, which shows the levels of individual frequencies. In most practical cases it is more convenient to use the sonogram, which shows the frequencies in time, but provides a lower level resolution as the levels are differentiated by color. The spectrum analyzer also provides a micro-sonogram (shown in the bottom of the panel) which uses the same color-based view as the sonogram.

### Fill

#### Fill

Fill button enables or disables the full-sized analyzer micro-sonogram. This means that the micro-sonogram at the bottom of the equalizer graph will fill the whole analyzer view. Color differentiation is often easier to understand than the classical spectrum analyzer, so this might help you better understand the spectrum of your audio material.

An alternative is to use the spectrum sonogram.



## Analyzer Rainbow Colors

Analyzer Rainbow Colors lets you see the analyzed sound spectrum in beautiful colors, following the same style as visible light. It ranges from infra-red colors for the lowest frequencies to ultra-violet colors for the highest frequencies in the analyzed audio. If rainbow colors are disabled, the analyzer and graph will be single-colored, following the setup from Settings/Graphs.

### Sonogram

## Sonogram

Sonogram button enables or disables the spectrum sonogram, which shows levels of individual frequencies in time. Levels are differentiated by color, so the accuracy is not as good as when using the spectrum analyzer. However, the time axis improves the visual orientation in the spectrum for typical audio signals. In contrast, the spectrum analyzer is more of a scientific tool.

### Settings

## Settings

Settings button shows the settings of the spectrum analyzer and the spectrum sonogram.



## Pause

Pause button stops the analyzer temporarily.



## Normalize

Normalize button enables or disables the visual normalization, which makes the loudest frequency be displayed at the top of the analyzer area (0dB); it does not normalise the sound. This is very useful for comparing frequency levels, however it does hide the actual level.

When comparing 2 spectrums you are usually interested mainly in the frequency level differences. In most cases both audio materials will have different overall levels, which would mean that one of the graphs would be "lower" than the other, making the comparison quite difficult. Normalize fixes this and makes the most prominent frequencies of the spectrum reach the top of the analyzer area (or have the most highlighted color in case of sonogram).



## Reset

Reset button resets analyzer graphs. This is particularly useful when analyzing infinite average and maximum values.



## Copy analysis

Copy analysis button copies the current state of the analysis into the system clipboard so that you can paste it into another analyzer for comparison. Hold **ctrl** to export the analysis into a CSV file.



## Paste

Paste button pastes the analysis from the system clipboard and displays it as the comparison in the graph.

## Band settings window



Band settings window contains settings for the particular band and can be displayed by right-clicking on a band or from a band list (if provided). On the left side you can see list of available filters, click on one to select it. On the right side, additional options and features are available.



## 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.

## General panel

GENERAL

Invert gain Swap gains ?

FREQUENCY

208.8 Hz

Q

0.71

GAIN

-15.78 dB

Slope

12345678910

Channels

LeftLeft + RightRight

General panel contains standard filter settings such as frequency or Q. Most of these values are available directly from the band graph, but it may be necessary to use these controls for more accurate or textual access.

**Invert gain**  
Invert gain inverts the gain of the band, e.g. makes -6dB from +6dB.

**Swap gains**  
Swap gains button swaps values between gain and dynamics gain.

**FREQUENCY**  
208.8 Hz  
**Frequency**  
Frequency defines the band's central frequency, which has different meaning depending of filter type.

**Q**  
0.71  
**Q**  
Q defines bandwidth. Please note that Q is an engineering term and the higher it is, the lower the bandwidth. Our implementation is trying to be more user-friendly, and by increasing the value (thus to the right), the bandwidth is increased as well. The editor still displays the Q value correctly.

**GAIN**  
-15.78 dB  
**Gain**  
Gain defines how the particular frequencies are amplified or attenuated. This parameter is used only by peak and shelf filters.

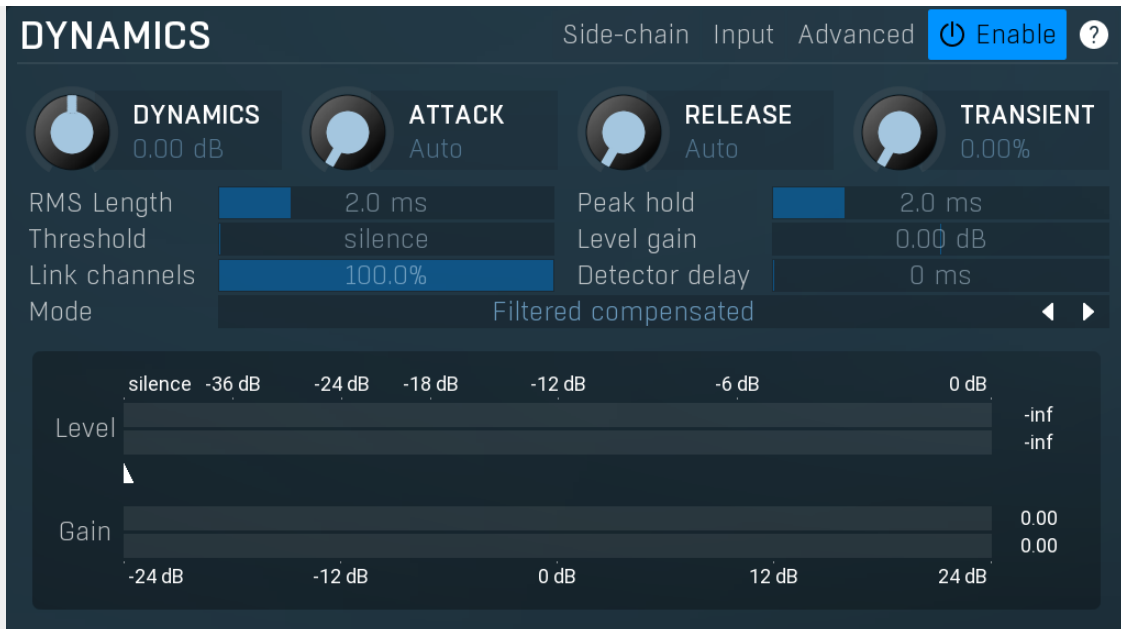
**Slope** 1 2 3 4 5 6 7 8 9 10 **Slope**  
Slope can potentially duplicate some of the filters creating steeper ones. By default, the slope is 1 and this usually means 2-pole 12 dB/octave filters. By specifying 2 you can make the plugin uses 4-pole 24 dB/octave filters instead etc. To see the actual slope of each filter look into the filter type list on the left.

Channels Left Left + Right Right  
**Channels**  
Channels controls which channels the band processes. If the input is stereo (left and right channels, L+R, selected on the toolbar **Channel mode** button), then you can make a band process only the left, only the right, or both channels. Similarly when the plugin is set to M/S channel mode, you can choose between mid, side or both channels.

When one of more bands are set to process a single channel, then 2 EQ curves are displayed, in red for the Left or Mid and in green for the Right or Side. If these are not distinct, then we recommend using a style with a light background for these graphs.

You cannot process left with one band and side with the other, because these are working in different encoding modes. In this case you can easily use 2 instances of the plugin in series, one in L/R mode and the other in M/S.

Dynamics panel



Dynamics panel contains settings of the dynamics processing which control how the filter behaves depending on input signal. Normal filters are static, meaning they don't change any features depending on the input signal. If you enable dynamic properties, by making the **dynamic gain** nonzero, the filter will start listening to the level of the input signal. This requires more CPU of course, as such a band is essentially an extremely complex generalized compressor, but the algorithms used are as efficient as it is technically possible.

A dynamic band varies the gain according to the input level. It can listen to the whole spectrum or to just part of it. By default it is driven by the partial spectrum, which it modifies itself, so, for example, when you have a high shelf, it is essentially listening to a high part of the spectrum. You can do many things with such a dynamic processor, but essentially it can work as a compressor or expander. There are many more advanced ideas that you can do and the full power hasn't really been explored yet.

#### Side-chain

### Sidechain

Sidechain switch makes the band measure the input level from the sidechain instead of the input it is processing. This can be used for various techniques, such as avoiding conflicts between bass and bass drum.

#### Input

### Input

Input switch makes the band measure the input level instead of current level in the chain of bands. When this is disabled (default) and the equalizer is processing the bands serially, which means that each band is processing the output from the previous stage, including level measurement. If you enable this switch however, the dynamic processing will be driven by the original input signal instead.

Please note that when **Side-chain** is on, this switch has no meaning, since side-chain has priority.

#### Advanced

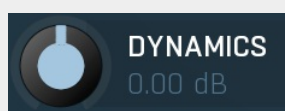
### Advanced

Advanced button displays additional settings for this band. These contain some more esoteric features, such as a dynamic transformation shape.

#### Enable

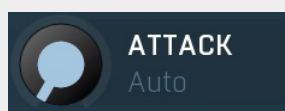
### Enable

Enable button enables the dynamic processing. You can use it to switch between enabled and disabled dynamic processing to check the differences.



### Dynamics

Dynamics defines the maximum gain of the filter that could be caused by the input signal. For example, if you set it to -24dB and the input signal contained in the band were very strong, the band will be set to an additional -24dB. This would work similarly to a compressor in that band.



### Attack

Attack defines the attack time, that is how quickly the level detector increases the measured input level. When the input peak level is higher than the current level measured by the detector, the detector moves into the attack mode, in which the measured level is increased depending on the input signal. The higher the input signal, or the shorter the attack time, the faster the measured level rises. Once the measured level exceeds the **Threshold** then the dynamics processing (compression, limiting, gating) will start.

There must be a reasonable balance between attack and **release** times. If the attack is too long compared to the release,

the detector will tend to keep the measured level low, because the release would cause that level to fall too quickly. In most cases you may expect the attack time to be shorter than the release time.

To understand the working of a level detector, it is best to cover the typical cases:

*In a **compressor** the attack time controls how quickly the measured level moves above the threshold and the processor begins compressing. As a result, a very short attack time will compress even the beginning transient of a snare drum for example, hence it would remove the punch. With a very long attack time the measured level may not even reach the threshold, so the compressor may not do anything.*

*In a **limiter** the attack becomes a very sensitive control, defining how much of the signal is limited and how much of it becomes saturated/clipped. If the attack time is very short, limiting starts very quickly and the limiter catches most peaks itself and reduces them, providing lower distortion, but can cause pumping. On the other hand, a higher attack setting (typically above 1ms) will let most peaks through the limiter to the subsequent in-built clipper or saturator, which causes more distortion of the initial transient, but less pumping.*

*In a **gate** the situation is similar to a compressor - the attack time controls how quickly the measured level can rise above the threshold at which point the gate opens. In this case you will usually need very low attack times, so that the gate reacts quickly enough. The inevitable distortion can then be avoided using look-ahead and hold parameters.*

*In a modulator, the detector is driving other parameters, a filter cut-off frequency for example, and the situation really depends on the target. If you want the detector to react quickly on the input level rising, use a shorter attack time; if you want it to follow the flow of the input signal slowly, use longer attack and release times.*



### Release

Release defines the release time, that is how quickly the level detector decreases the measured input level. The shorter the release time, the faster the response is. Once the attack stage has been completed, when the input peak level is lower than the current level measured by the detector, the detector moves into the release mode, in which the measured level is decreased depending on the input signal. The lower the input signal, or the shorter the release time, the faster the measured level drops. Once the measured level falls under the **Threshold** then the dynamics processing (compression, limiting, gating) will stop.

There must be a reasonable balance between **attack** and release times. If the attack is too long compared to release, the detector would tend to keep the level low, because release would cause the level to fall too quickly. Hence in most cases you may expect the attack time to be shorter than the release time.

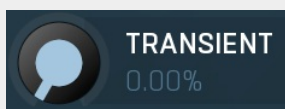
To understand the working of a level detector, it is best to cover the typical cases:

*In a **compressor** the release time controls how quickly the measured level falls below the threshold and the compression stops. As a result a very short release time makes the compressor stop quickly, for example, leaving the sustain of a snare drum intact. On the other hand, a very long release keeps the compression working longer, hence it is useful to stabilize the levels.*

*In a **limiter** the release time keeps the measured level above the limiter threshold causing the gain reduction. Having a very long release time in this case doesn't make sense as the limiter would be working continuously and the effect would be more or less the same as simply decreasing the input gain manually. However too short a release time lets the limiter stop too quickly, which usually causes distortion as the peaks through the limiter to the subsequent in-built clipper or saturator. Hence release time is used to avoid distortion at the expense of decreasing the output level.*

*In a **gate** the situation is similar to a compressor - the release time controls how quickly the measured level can fall below the threshold at which point the gate closes. Having a longer release time in a gate is a perfectly acceptable option. The release time will basically control how much of the sound's sustain will pass.*

*In a modulator, the detector is driving other parameters, a filter cut-off frequency for example, and the situation really depends on the target. If you want the detector to react quickly on the input level falling, use a shorter release time; if you want it to follow the flow of the input signal slowly, use longer attack and release times.*



### Transient

Transient lets you mix the level follower output with a transient detector output. This lets you follow signal level, transients or both. Note that since transient level is usually lower than level detector's output, **Level gain** is only applied on the level detector's signal, so you can use this to compensate for the difference in level.

**RMS Length** 2.0 ms **RMS length**

RMS length smoothes out the values of the input levels (not the input itself), such that the level detector receives the pre-processed signal without so many fluctuations. When set to its minimum value the detector becomes a so-called "peak detector", otherwise it is an "RMS detector".

When you look at a typical waveform in any editor, you can see that the signal is constantly changing and contains various transient bursts and separate peaks. This is especially noticeable with rhythmical signals, such as drums. Trying to imagine how a typical attack/release detector works with such a wild signal may be complex, at least. RMS essentially takes the



surrounding samples and averages them. The result is a much smoother signal with fewer individual peaks and short noise bursts.

RMS length controls how many samples are taken to calculate the average. It stabilizes the levels, but it also causes a slower response time. As such it is great for mastering, when you want to lower the dynamic range in a very subtle way without any instabilities. However, it is not really desirable for processing drums, for example, where the transient bursts may actually be individual drum hits, hence it is usually recommended to use peak detectors for percussive instruments.

Note that the RMS detector has 2 modes - a simplified approximation is used by default, and a true RMS is processor can be enabled from the advanced settings (if provided). Both respond differently, neither of them is better than the other, they are simply different.

### Peak hold 2.0 ms Peak hold

Peak hold defines the time that signal level detector holds its maximum before the release stage is allowed to start. As an example, you can imagine that when an attack stage ends there can be an additional peak hold stage and the level is not yet falling, before the release stage starts. This is true only when **true peak** mode is enabled (check the advanced detector settings if available).

It is often used in **gates** to avoid the gated level falling below the threshold too quickly, while having short release times. If you want the gate to close quickly, you need a short release time. But in that case the ending may be too abrupt and even cause some distortion. So you use the peak hold to delay the release stage.

It is also used along with **look-ahead** to avoid distortion in **limiters and compressors**. If you need a very short attack, the attack stage may be too quick and cause distortions. In limiters this attack time is often 0ms, in which case it becomes a clipper. Setting look-ahead and peak hold to the same value will make the detector move ahead in time, so that it can react to attack stages before they actually occur and yet hold the levels for the actual signal to come.

### Threshold silence Threshold

Threshold controls the minimum level above which the dynamic gain actually starts working.

### Level gain 0.00 dB Level gain

Level gain controls the gain applied to the detector, which can be used for example when the input level is too low, so that dynamic processing would be negligible, unless the level is boosted.

### Link channels 100.0% Link channels

Link channels controls how much the signal level for each channel is controlled by the other channels. With 0% the link is disabled and each channel is not affected by the other channels at all. This is suitable to balance stereo channels, for example. With 100% the link is enabled and all channels are controlled by levels of all channels equally (that is the average level of those channels), therefore the processor will apply the same amount of processing on all channels. This is the default in most cases as it preserves relative levels between the channels.

### Detector delay 0 ms Detector delay

Detector delay lets you delay the detector input, hence the band will react later than the actual input signal.

### Mode Filtered compensated Mode

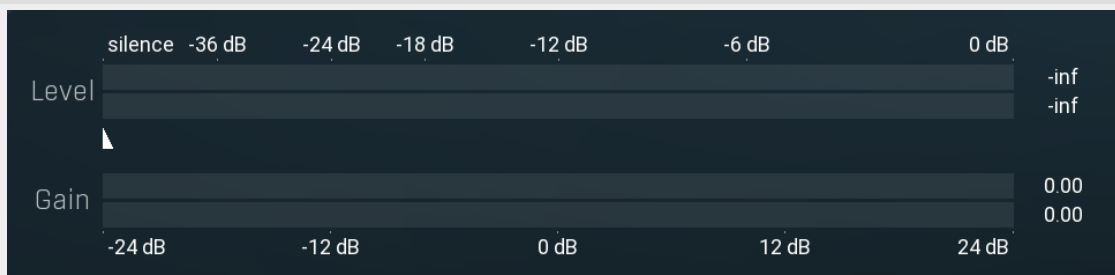
Mode controls the way the band reacts to the input signal. It has no meaning if the dynamic gain is 0dB.

**Filtered compensated** mode is default and it means that the source for measuring input level is a filtered signal with additional compensation. For example, when using a low-shelf filter, the signal is low-passed with a filter with the same settings as the low-shelf, therefore the low-shelf filter is affected only by the signal the low-shelf is actually amplifying or attenuating. Since a low-passed signal with cut-off at 100Hz has usually a much lower level than the one filtered with cut-off at 10 kHz, additional compensation is performed to diminish these differences.

**Filtered** mode is similar, but the compensation is not performed. This may be advantageous for audio materials that do not contain the full spectrum, e.g. a bass line, where the compensation may make things complicated.

**Entire spectrum** mode is the simplest - it simply takes the input signal without any further processing. This may be useful for example to attenuate selected frequencies when the input level gets too high.

## meters



### Threshold

Threshold controls minimum level at which the dynamic gain actually starts working.

## Harmonics panel

## HARMONICS

Linear Dynamics by fundamental ?



HARMONICS  
0.00%



SEMITONES  
12.00



MAXIMAL COUNT  
16

Harmonics



Harmonics panel contains parameters of the harmonics - clones of the main band created at higher frequencies derived from the frequency of the main band. This is often useful for removing natural noises, which usually bring some harmonics with them etc.

Linear

### Linear

Linear button enables the linear harmonics spacing. When the main band frequency is say 100Hz and the **Semitones** value is 12, then in the default logarithmic mode the harmonics are 200Hz, 400Hz, 800Hz etc., increasing by 12 semitones (1 octave) each time. This is suitable because the filters themselves are logarithmic.

However harmonics generated by physical instruments are not spaced in this way. Rather, for a **Semitones** value of 12, they increase by a multiple of 12/12 of the main frequency each time. For example, for a base frequency of 100Hz, they will be at 200Hz, 300Hz, 400Hz, 500Hz etc. In linear mode the harmonics work in this way, but please note that then there is only a limited set of harmonics and Q is modified to approximate a reasonable behaviour, which is not always possible.

Dynamics by fundamental

### Dynamics by fundamental

Dynamics by fundamental switch causes each harmonic to be driven by the same detector settings as set for the main band. It is disabled by default, which means that each harmonic is literally a clone of the original filter and has its own dynamics detector depending on its own frequency.

Please note that if you want each harmonic to behave in exactly the same way as the main band, you also need to switch on the Input (at the top of the Dynamics panel), otherwise the harmonics would be measuring the signal processed by the main band.



HARMONICS  
0.00%

### Harmonics

Harmonics defines the gain of the created harmonics. With maximum value (+/- 100%), all harmonics will have the same gain as the main band. A lower value makes the higher harmonics have lower gain. A negative depth will make alternate harmonics have positive and negative gains and is particularly useful for creative effects.



SEMITONES  
12.00

### Semitones

Semitones defines the frequency interval of the harmonics. For example, if the band is at 100Hz and the number of semitones is 12 (default), then the first harmonic will be at 200Hz (12 semitones higher), second at 400Hz etc., increasing by 12 semitones (1 octave) each time. Thus they are logarithmically-spaced harmonics. When linearly-spaced harmonics are enabled, this merely changes the ratio between them. In this mode, 100Hz is followed by 200Hz, 300Hz, 400Hz, 500Hz etc, that is, increasing by a multiple of 12/12 of the main frequency each time.

For a value of 7 (a perfect fifth), the logarithmic harmonics would be at 150Hz, 225Hz, 337.5Hz, 506.25Hz etc, increasing by 7 semitones (= 50%, as  $1.05946^7 = 1.498$ ) each time and the linear harmonics would be at 158Hz, 251Hz, 397Hz, 628Hz etc, increasing by 7/12 each time.



MAXIMAL COUNT  
16

### Maximal count

Maximal count defines the maximum number of harmonics that could be created. The harmonics that are created depends on them being activated in the **Harmonics grid**.

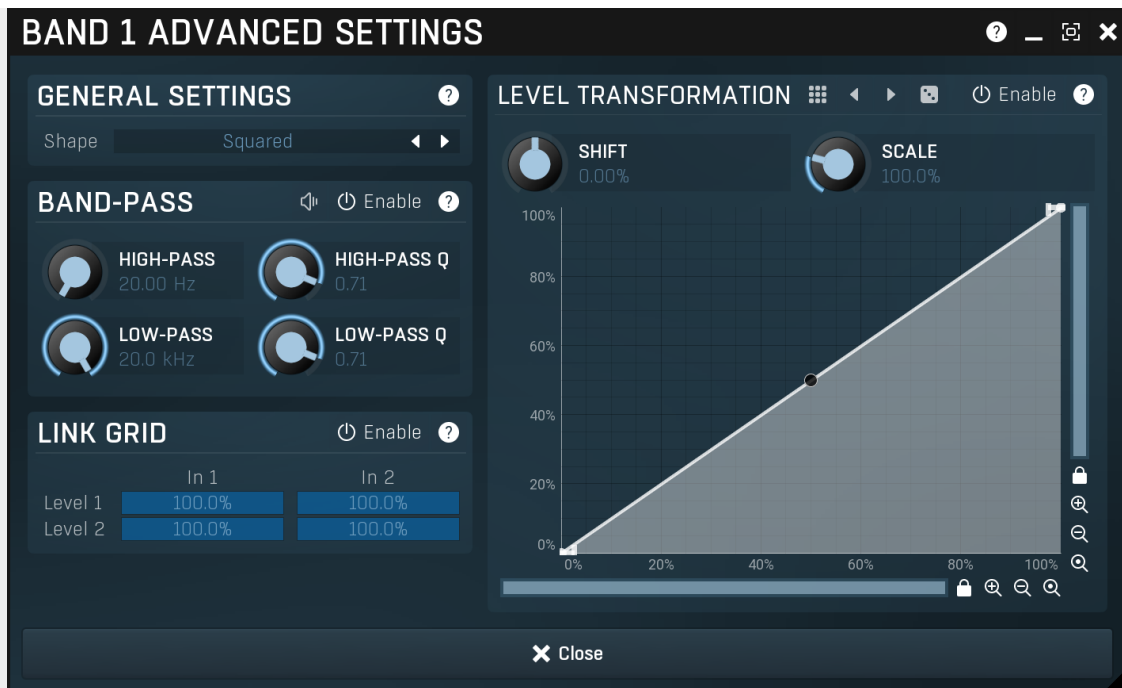
## Harmonics grid

Harmonics



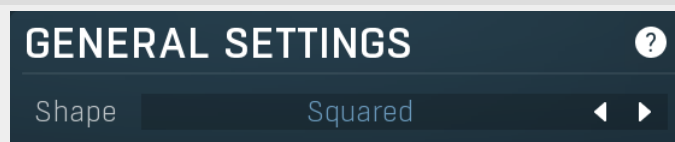
Harmonics grid is useful to turn on/off particular harmonics manually. Click any one to enable / disable it.

## Band advanced settings



Band advanced settings contains additional settings for the band. These contain some more esoteric features, such as a dynamic transformation shape. It can be displayed by clicking the right mouse button on a band while holding **Ctrl**, from the basic band settings window, or from the band list if provided.

## General settings panel



General settings panel contains additional parameters, which are too scientific to be available from the main band settings.



Shape affects the processing shape. The plug-in features specific non-linear transfer shapes which affect the way the level are interpreted. **Logarithmic** mode is the most physical one, increase from, say, -90dB to -80dB and from -10dB to 0dB produces the same difference in the output dynamic gain. However from the nature of it it tends to generate high gains and usually setting a threshold is needed. **Linear** mode on the other hand tends to stay near minimum gains and usually is the most aggressive. **Squared** mode is a compromise between these two. Comparing the three modes, Linear mode requires the least amount of CPU power and Logarithmic requires the most.

## Band-pass panel



Band-pass panel contains parameters of the band pass, which you can use to process the signal that is used measure level of the band additionally. For example, you may want a band at high frequencies to react to bass content; you can do this by placing the band anywhere on the high frequencies and set the low-pass at say 200Hz.



**Play**

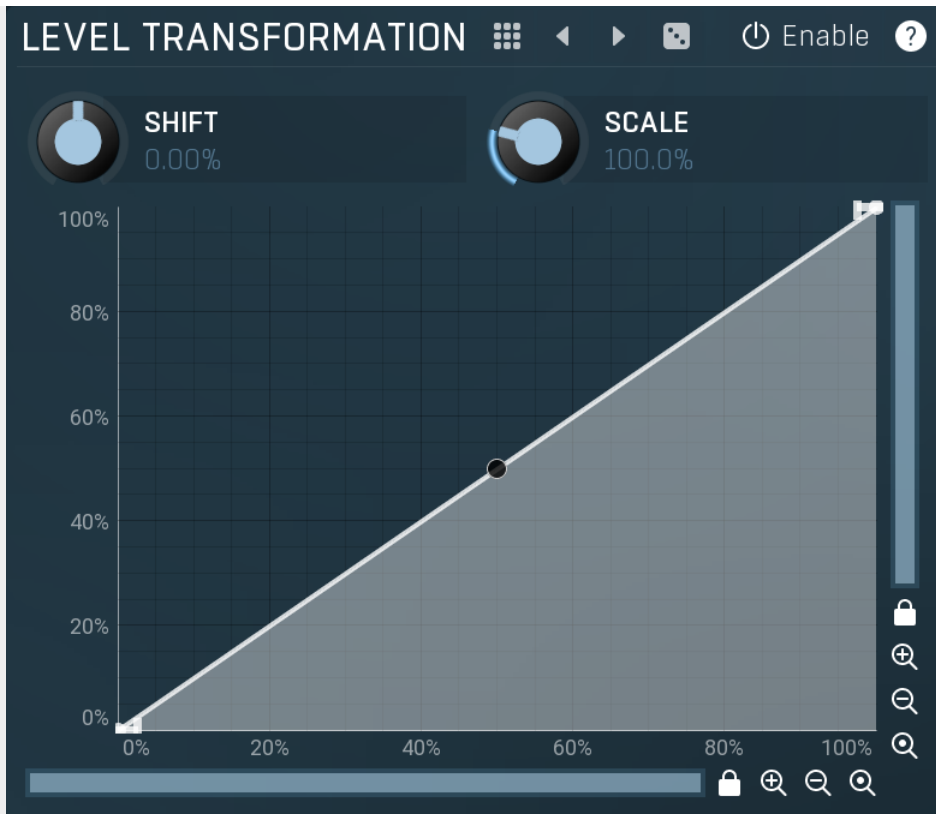
Play button enables the band-pass monitoring and hence could be useful to tweak the band pass.



**Enable**

Enable button enables the band-pass module. It is off by default to save CPU resources.

## Level transformation



Level transformation graph lets you transform the dynamic gain according to the input level. The X axis contains the input level; the Y axis controls the output level, which is then used to set the dynamic gain.



### 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.



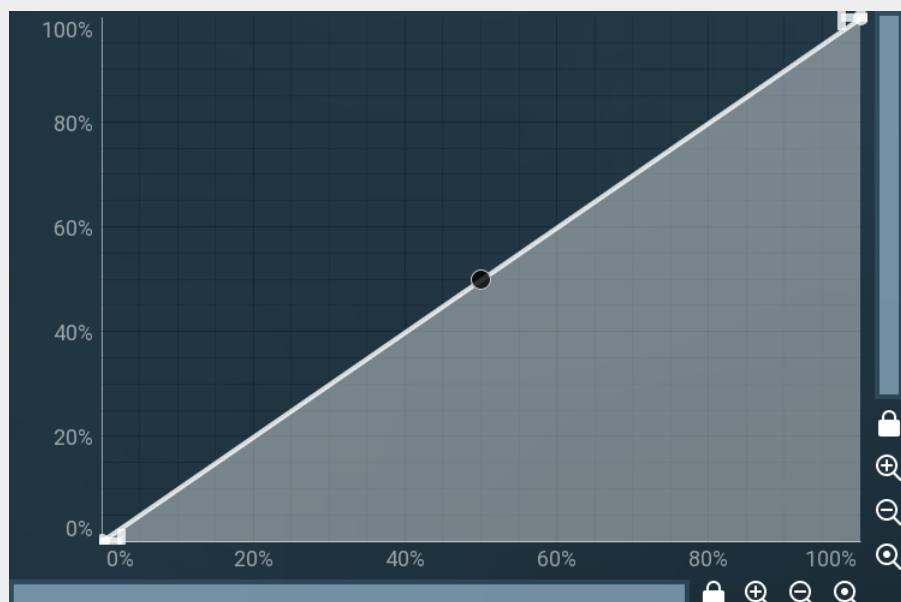
### Randomize

Randomize button loads a random preset.



### Enable

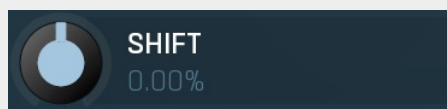
Enable button enables the level transformation module. It is off by default to save CPU resources.



## Envelope graph

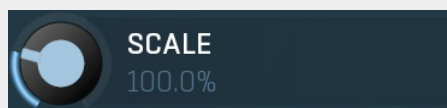
Envelope graph provides an extremely advanced way to edit any kind of shape that you can imagine. An envelope has a potentially unlimited number of points, connected by several types of curves with adjustable curvature (drag the dot in the middle of each arc) and the surroundings of each point can also 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.



### Shift

Shift lets you virtually shift the whole graph vertically. This basically shifts the dynamic gain.



### Scale

Scale lets you virtually scale the whole graph vertically. This basically scales the dynamic gain.

## Link grid panel

LINK GRID			Enable ?
	In 1	In 2	
Level 1	100.0%	100.0%	
Level 2	100.0%	100.0%	

Link grid panel controls the linking between the channels; that is. how the input level in each channel affects the levels in the other channels. By default the way channels affect processing in other channels depends solely on the **Link channels** parameter.

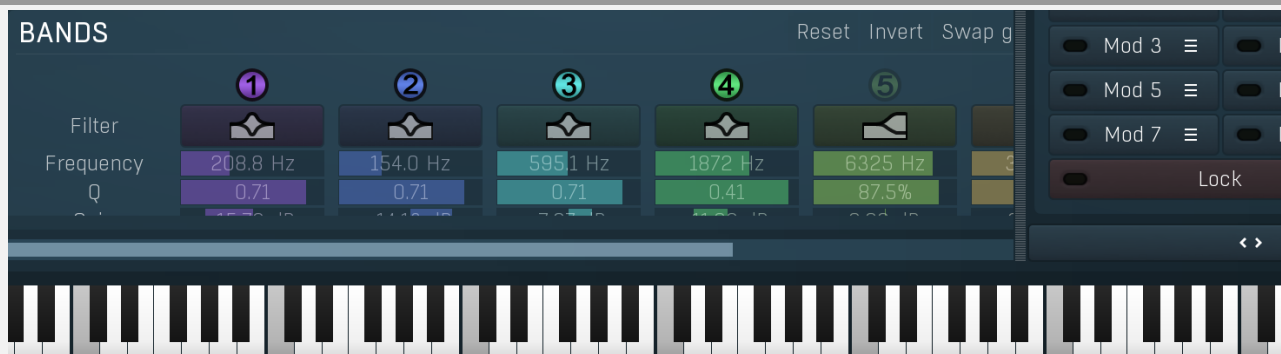
Here you can set up a more complicated relationship. For example, you can make the left channel (1) respond to the right channel (2) only and vice versa. Each column in the grid is an input and each row is an output. Each output level is a mix of the factored input levels. For that example above, the values for "Level 1" would be 0% and 100%, and for "Level 2" they would be 100% and 0%.



### Enable

Enable button enables the link-grid module. It is off by default to save CPU resources.

## Bands panel



Bands panel contains the list of available bands along with their basic parameters. You can use it to enable/disable a band, change the parameters and show the band settings window if you do not wish to edit the bands within the equalizer graph panel or if you need to set some values by numeric text entry. The panel is collapsed by default, as it can take a lot of space.



### Reset

Reset button restores the original equalizer settings.



### Invert

Invert button inverts the gains of all bands.



### Swap gains

Swap gains button swaps values between gain and dynamics gain.

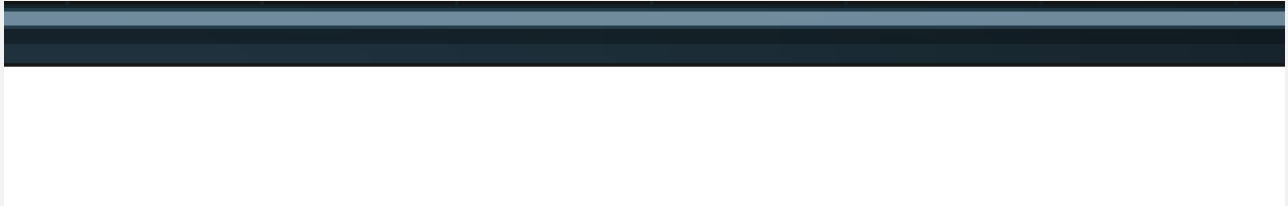


Mod 3

### Band graphs

Band graphs button enables/disables display of band graphs.

## Automatic equalizer panel



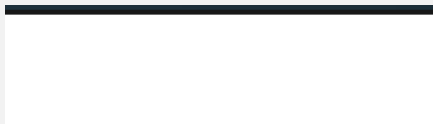
Automatic equalizer panel contains long-term analysis and automatic equalization functions.



### Analyze source

Analyze source button starts or stops the source analysis, source defines how you want your audio to sound. In your host, route the source audio only to the plug-in and start playback, then press this button to start the analysis. When the graph (shown as a **red line**) stops moving the analysis is finished and you can press the button again to complete the process.

Alternatively, you can analyze an audio file (WAV, MP3, FLAC etc.) offline by clicking the **File** button and browsing to the file or by dragging & dropping the file from your host or Explorer / Finder onto the **Analyze source** button (this latter method may not work in all hosts, especially on Mac). Three other 3 buttons let you save and load an analysis or even draw the desired response manually.

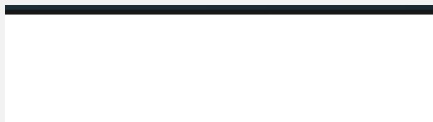


### Analyze target

Analyze target button starts or stops the target analysis, target refers to the audio that you want to process. In your host, route the target audio only to the plug-in and start playback, then press this button to start the analysis. When the graph (shown as a **green line**) stops moving the analysis is finished and you can press the button again to complete the process.

Alternatively, you can analyze an audio file (WAV, MP3, FLAC etc.) offline by dragging & dropping the file from your host or Explorer / Finder onto the **Analyze target** button (this may not work in all hosts, especially on Mac).

This tip may come in handy - instead of playing the whole song back to get an accurate analysis you can render the song and analyse the output file, which will probably be faster.



### Equalize

Equalize button performs automatic equalization - it adjusts the bands to match the source and target analyses as closely as possible. To do that, you need to have the analyses of both the source and target audio first. When you have both analyses ready, you will see the **red** (source) and **green** (target) analyses in the graph area and this button becomes available.

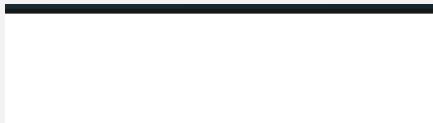
First get the source analysis using the **Analyse source** button. The Source is the reference audio material that you want your track to sound like.

This is most likely a different track, so you have several options to get this analysis. You can move the plugin to the reference track, perform the analysis then move the plugin onto the track you want to process. Or, open another instance of the plugin on the reference track, perform the analysis and copy the analysis (using the copy/paste buttons, below the A-H preset selectors and A|B comparison button) to the plugin on your track. Or, by saving and loading the analysis (using the **Save** and **Load** buttons). Alternatively, you can draw the desired spectrum or analyse an WAV/MP3/FLAC file by dragging & dropping the file onto the **Analyze source** button.

Secondly, use **Analyse target** button to analyse the audio that you are processing (the **green** line in the graph).

Finally press the **Equalize** button to perform the equalization.

Automatic equalizer tries to match the spectral content of the source analysis graph to the target analysis, producing an equalization curve that aims to make the target audio sound tonally more like the source audio. In most cases the result will be too strong so it is worth considering lowering the **Dry/Wet** parameter to say 30-40% to get a more natural output. You can also make use of the **Smoothness** parameter, above the graph area. Increasing the smoothness before pressing **Equalize** will create a less-pronounced equalization curve.



### Separate

Separate button performs automatic separation - adjusting the bands so that the target does not contain those frequencies that are prominent in the source. This is useful, for example, during mixing to avoid collisions between multiple tracks. To do that, you need to have analyses of both the source and target audio first. When you have both analyses ready, you will see **red** (source) and **green** (target) analyses and this button becomes available.

*For example, say you want to avoid collisions between bass and bass drum. One of them will have to be sacrificed and processed by the equalizer, let's choose the bass. In that case you would analyse the bass drum as the source (the **red** line) and put the equalizer into the bass track afterwards and analyse that as the target (the **green** line). **Separate** would then produce an equalization curve that reduces, in the bass track, those frequencies that are prominent in the bass drum track.*

*Another example is typical when your mix is already busy, but you need to put one more track to it. In this case you analyse the whole mix as the source and your new track as the target. The separation will then allow only those frequencies from your new track that are not prominent in the whole mix; in other words, frequencies that are already prominent in the mix are not affected by those same frequencies in the new track. If this separation were not done, then those frequencies would start colliding with the rest of your mix and that could make it sound muddy and crowded.*

When you have both analyses ready, click this button to perform the separation. In most cases the result will be too strong so it is worth considering lowering the **Dry/Wet** parameter to say 30-40% to get more natural output. You can also make use of the **Smoothness** parameter, above the graph area. Increasing the smoothness before pressing **Equalize** will create a less-pronounced equalization curve.



### Reset

Reset button clears both the source and target analyses.



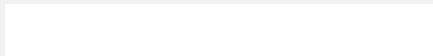
### Remove resonances

Remove resonances button takes the target analysis, tracks peaks in it and creates notch filters to remove them. It is especially useful with various audio materials, such as drums, where resonances of the instruments are too prominent. You simply analyze the part of the audio material, where the resonances are mostly audible, and click this button. Then you can just disable the bands that remove the frequencies that you actually want to keep. It may also be worth trying to play with the Q values of each band to make sure the resonances are completely gone.



### Swap source and target

Swap source and target button swaps the analyses of source and target and can be helpful when you want to try equalizing 'the other way around'.

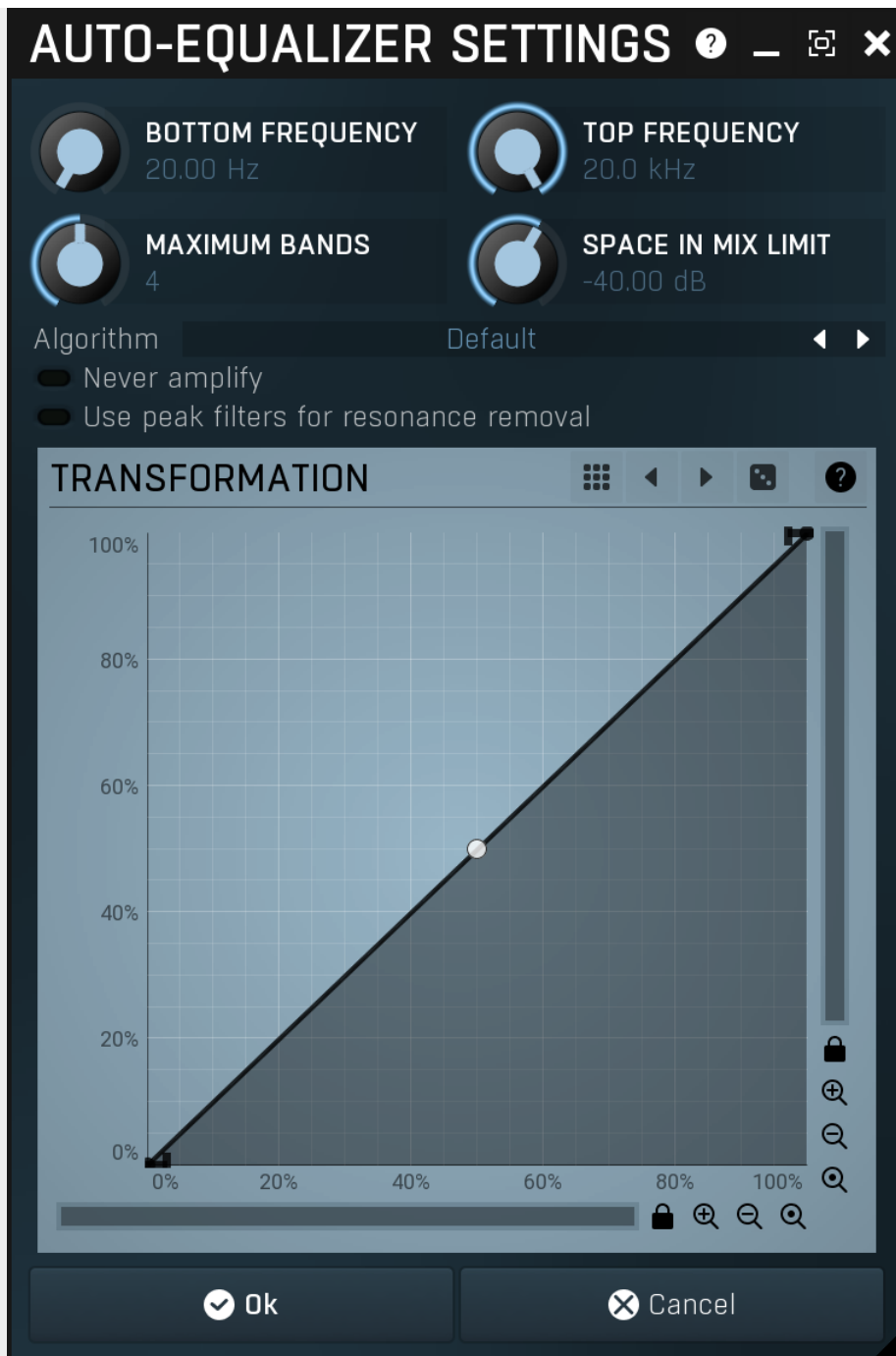


### Settings

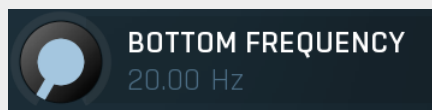
Settings button shows additional settings of the automatic equalizer algorithm.

## Auto-equalizer settings





Auto-equalizer settings provides additional settings for the automatic equalization algorithm.



#### Bottom frequency

Bottom frequency defines lowest frequency taken into account when performing automatic equalization. Use it to avoid unnecessary processing of the bottom-end.



#### Top frequency

Top frequency defines highest frequency taken into account when performing automatic equalization. Use it to avoid unnecessary processing of the high-end.



#### Maximum bands

Maximum bands defines maximal number of bands that will be configured during the automatic equalization. Generally more bands provide more accurate match to the source analysis, however this may not always be desired. You can also make use of the **Smoothness** parameter, above the graph area. Increasing the smoothness before pressing **Equalize** will create a less-pronounced equalization curve.



## SPACE IN MIX LIMIT

-40.00 dB

### Space in mix limit

Space in mix limit controls the minimal level below which a particular frequency is considered silent. It is used for the **Separate** feature to determine which frequencies in the source (full mix for example) are candidates for separation and which are not.

For example, if this value is -20dB and the level of the frequency 1000Hz in the source analysis is -20dB, then the separation engine will consider 1000Hz to be "available" in the target audio and will not try to remove this frequency from the target. However if the level in the source were -10dB, then the engine would consider this frequency 50% occupied and may try to use a filter to remove some of the 1000Hz from the target if it contains this frequency as well.

In other words, frequencies in the source with levels louder than the mix limit will be those that would be reduced in the target.

### Algorithm

Default

### Algorithm

Algorithm selector lets you choose which method is used to convert the frequency response into a configuration of the equalizer bands. (This algorithm itself is actually very complex and requires lots of CPU when adjusting the bands during the equalization/separation processes.) Multiple algorithms are available, each of them with a slightly different output.

**Default** provides the most accurate results, but it takes lots of CPU. Note, as mentioned above, that this CPU usage is required only for the pre-computing, the actual audio processing requirements are the same for all algorithms.

**Optimized** algorithm is much faster, however it can produce slightly less accurate results. It may be more suitable because it smooths out the peaks in the frequency response.

**Super-fast** mode is the fastest of all of them, but it trades speed for accuracy in its results. You may find it useful in specific cases.



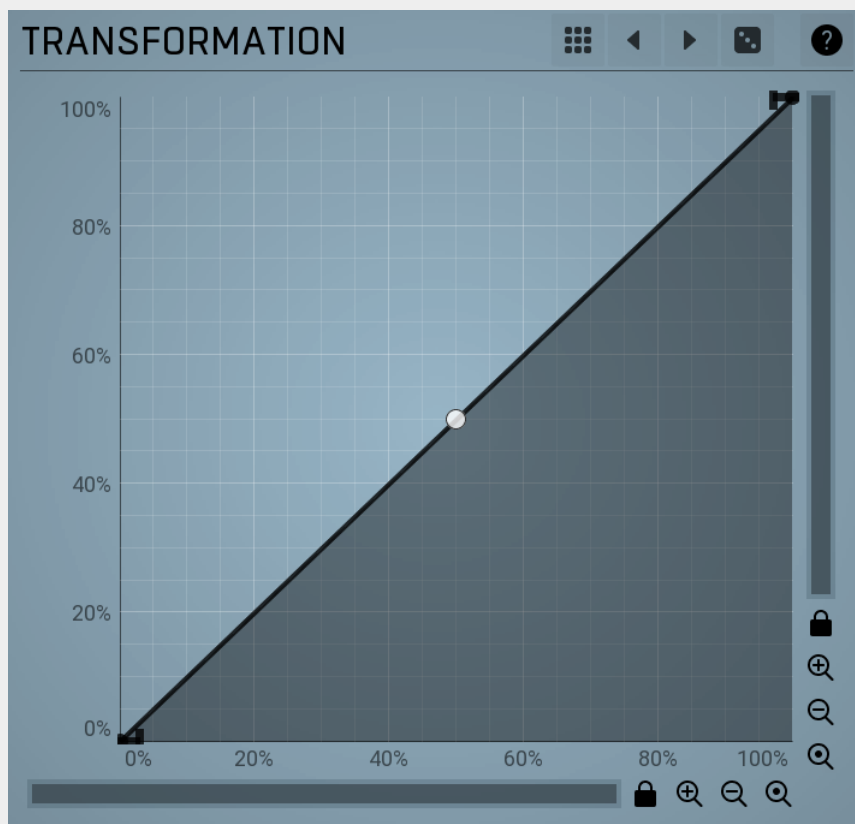
### Never amplify

Never amplify ensures that the automatic equalization only attenuates frequencies so that no amplification occurs. This may be desired, since amplification often increases noise level. On the other hand, due to its nature, it will probably lower the output level. It may be worth considering using AGC to set the output gain afterwards.



### Use peak filters for resonance removal

Use peak filters for resonance removal option makes the **Remove resonances** feature use peak filters instead of the default notch filters, which are more effective, but **Dry/Wet** doesn't affect them.



### Transformation

Transformation changes the requested frequency response. The engine first determines the optimum frequency response (the algorithms for equalization and separation are different). Then it computes the optimal response of the equalizer. Before it approximates this response using the equalizer filters, that response can be transformed. The transformation graph shows the original requested response on the X-axis and the new one on the Y-axis. For example you can exaggerate or sharpen the results just using the single point curvature.



### 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.



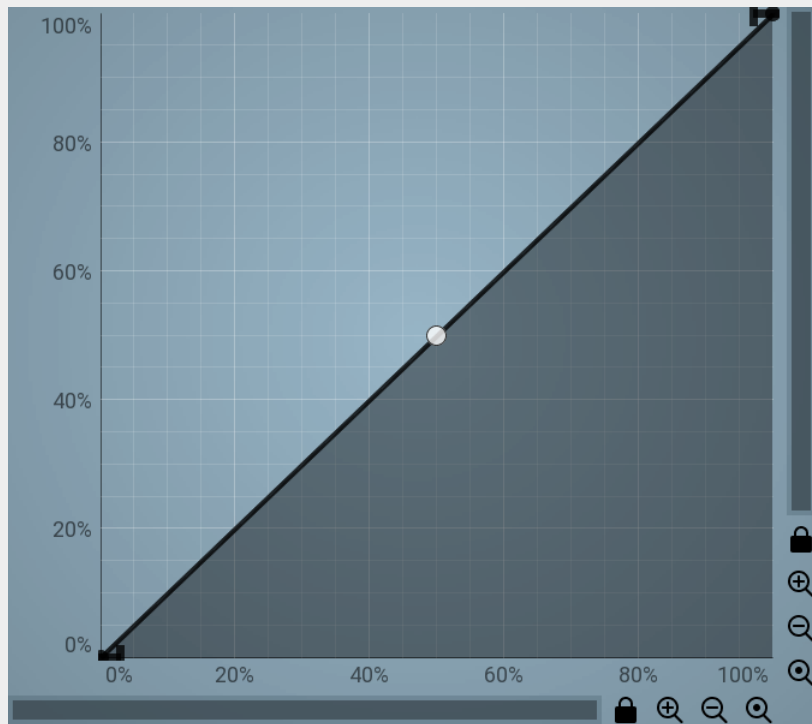
### Right arrow

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.



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



### 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.

## Voicing panel

## VOICING

Mono toggle

Allow multiple monophonic voices ?

Voice restart mode	Smooth	◀ ▶	Voice to kill	Last	◀ ▶
Max voices	128	◀ ▶	Max voices per note	1	◀ ▶
Stop voices when silent	Off		Lowest note	0 (C-1)	
Filter follow mono	10 ms		Highest note	127 (G9)	
Global ADSR use	Volume	◀ ▶	Sustain minimum	50.0%	
<input checked="" type="checkbox"/> Enable sustain pedal	<input checked="" type="checkbox"/> Enable sostenuto pedal		<input type="checkbox"/> Enable volume MIDI controller		
<input checked="" type="checkbox"/> Enable panorama MIDI controller	<input type="checkbox"/> Follow pitchbend after note-off		<input checked="" type="checkbox"/> Equal random every time		

Voicing panel contains some global settings controlling how new voices are created.

Mono toggle

### Mono toggle

Mono toggle switch is useful in monophonic modes and if enabled, releasing a new note while holding the previous one makes the pitch of the voice changed back to the note you are holding.

Allow multiple monophonic voices

### Allow multiple monophonic voices

Allow multiple monophonic voices switch is useful in monophonic modes and if enabled, multiple voices can actually occur at the same time. This has an effect for sounds with long release times. Holding a note and pressing another would change its pitch and do whatever you are used to in normal monophonic modes, but once you release all keys, the release stage starts. Then when you press another key while the previous voice is still in release stage, normally this voice is switched yet again to the new pitch. But if this switch is enabled, pressing a new key (while no other is held) creates a new voice leaving the previous one to decay until finished.

Voice restart mode

Smooth

◀ ▶

Voice restart mode

Voice restart mode is useful in monophonic modes and how the new notes are replacing the old ones.

**None** means that the pitch is change, but nothing else happens.

**Smooth** makes all envelopes restarted into the attack stage smoothly connecting them, hence avoiding abrupt changes resulting in clicks.

**Smooth even off** is similar, but the envelopes are restarted into the attack stage even when jumping back from notes being released as a result of **Mono toggle**

**Brutal** makes all envelopes restarted the initial position, hence causing abrupt changes, which may however be desired for certain sounds.

**Restart** mode creates new voices instead of changing the existing ones. 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 envelopes are followed and you can get interesting overlaps of the release stages.

**Restart (brutal)** is similar to Restart, but existing voice(s) are immediately brutally stopped.

Voice to kill

Last

◀ ▶

Voice to kill

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

Max voices

128

◀ ▶

Max voices

Max voices controls the maximum number of voices that can play at any moment. You can use this to avoid overloading your CPU when too many voices are played. By default the plugin provides unlimited number of voices.

Max voices per note

1

◀ ▶

Max voices per note

Max voices per note controls the maximum number of voices on the same note and channel, that can play at any moment in

**Polyphonic (single)** voicing mode. You can use this to avoid overloading your CPU when too many voices are played, but still allowing multiple of them, which may be necessary for realistic acoustic instruments such as piano for example.

Range: 1 to 20, default 1

Stop voices when silent

Off

Stop voices when silent

Stop voices when silent provides an alternative to the default stopping the voice based on the global ADSR. Normally each voice is stopped when the global ADSR ends, whether it is used to actually control the volume or not. When you enable this however, the voices will be terminated after a specified period of silence. So every time the voice is rendered, the output level is measured and when it is silent (below -120dB), a time measurement is performed and once the time reaches your requested value, the voice will be terminated.

Range: Off to 10000 ms, default Off

Lowest note

0 (C-1)

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

Filter follow mono

10 ms

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

Highest note

127 (G9)

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

Global ADSR use

Volume



Global ADSR use

Global ADSR use controls how the global ADSR is applied. Note that it is always used to determine the end of each voice, no matter what this mode is set to.

**Off** makes the engine use it only to detect the end of each note. You can use it manually for modulation however.

**Volume** is the default mode, which makes the engine apply it as volume level on each output of the generator.

**Volume squared** is similar, but makes the volume curve more drastic.

Sustain minimum

50.0%

Sustain minimum

Sustain minimum controls the minimum value of the sustain pedal MIDI controller to be considered 'open'. Above this value all notes being pressed are held until the sustain pedal is released. In some cases, such as simulating an actual piano, you may want to lower this value to simulate partial sustain.

Range: 0.00% to 100.0%, default 50.0%

☒ Enable sustain pedal

Enable sustain pedal

Enable sustain pedal makes the instrument listen to the sustain pedal when processing notes. If this is disabled, the instrument ignores the sustain pedal when processing notes.

☒ Enable sostenuto pedal

Enable sostenuto pedal

Enable sostenuto pedal makes the instrument listen to the sostenuto pedal when processing notes. If this is disabled, the instrument ignores the sostenuto pedal when processing notes.

☐ Enable volume MIDI controller

Enable volume MIDI controller

Enable volume MIDI controller makes the plugin automatically follow the volume MIDI messages. It is disabled by default, since it should be handled manually using global MIDI settings, which let you attach it to the easy screen multiparameters.

☒ Enable panorama MIDI controller

Enable panorama MIDI controller

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

☐ Follow pitchbend after note-off

Follow pitchbend after note-off

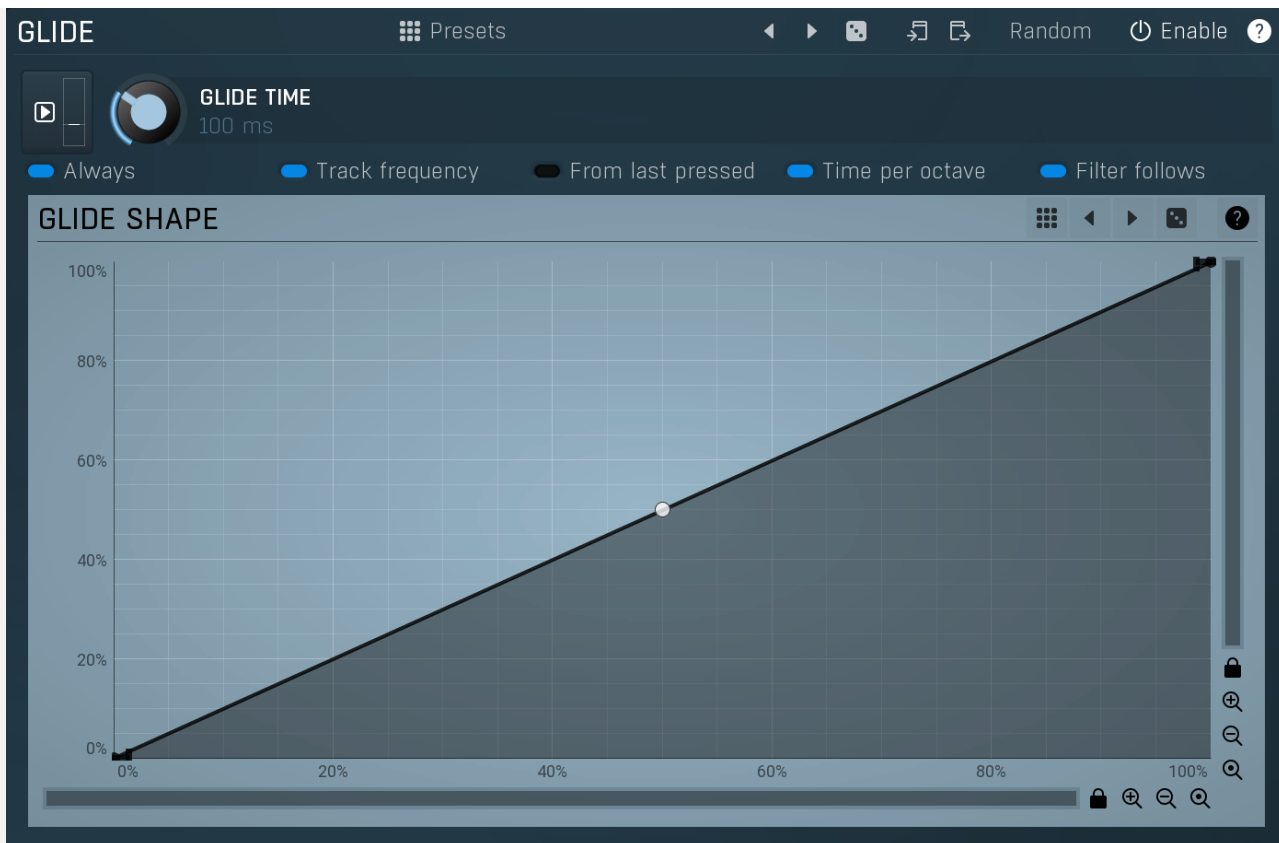
Follow pitchbend after note-off makes the plugin follow any changes to the pitchbend even after the key has been released and the voice is decaying in the release stage. Normally this is disabled, which is necessary for MPE to work properly.

☒ Equal random every time

Equal random every time

Equal random every time controls the way the many random things used in the plugin are initialized when rendering. By keeping this enabled you will get identical (in most cases) performance every time. By disabling this, the plugin will work the same way as most other plugins and hardware, which will sound slightly differently every time.

## Glide panel



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

Presets

## 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

Random button generates random settings using the existing presets.



Always

## 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

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**

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**

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.

## Filter follows **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**

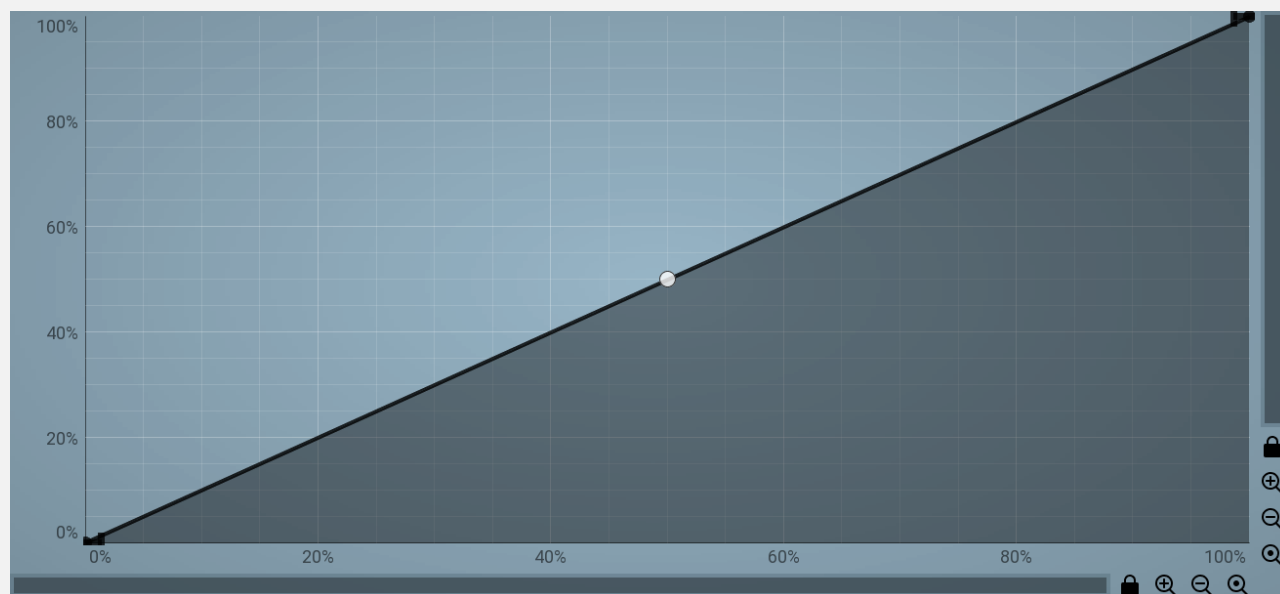
Left arrow button loads the previous preset.

## **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.

## Volume control panel

### VOLUME CONTROL



RANGE  
-30.00 dB

☒ Velocity interpolation  
☐ Follow velocity off

☒ Follow velocity  
☐ Follow aftertouch

Volume control panel contains controls defining how voice volume is calculated.



RANGE  
-30.00 dB

#### Velocity range

Velocity range controls the velocity to volume transformation. When set to minimum it becomes "Off" and the volume is calculated not logarithmically, but linearly. Any other value defines the gain of minimum velocity. For example, if you set this to -20dB, than notes with highest velocities will be about 20dB louder than notes with lowest velocities.

Range: Off to 0.00 dB, default -30.00 dB



Velocity interpolation

#### 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.



Follow velocity

#### Follow velocity

Follow velocity switch makes the plugin initialize the volume of each voice according to the velocity. This is desired on most controllers, but may not be working well with certain types of expression on 3D MPE controllers. It is then useful to disable this option and enable Follow aftertouch instead, which will make the instrument initialize a very small velocity and wait for aftertouch messages.



Follow velocity off

#### Follow velocity off

Follow velocity off switch makes the plugin change the volume of voices when a note off with a velocity specified. Some MIDI controllers (especially MPE controllers) send this value as a signal of 'how quickly the user released the key'. In many cases this is not desired though, so it is disabled by default.



Follow aftertouch

#### Follow aftertouch

Follow aftertouch switch makes the plugin listen to aftertouch MIDI messages and adjust the volume/velocity of each voice accordingly. Usually this is disabled and the aftertouch is only used to modulated some parameters providing the additional feel when playing the instrument.



### 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.



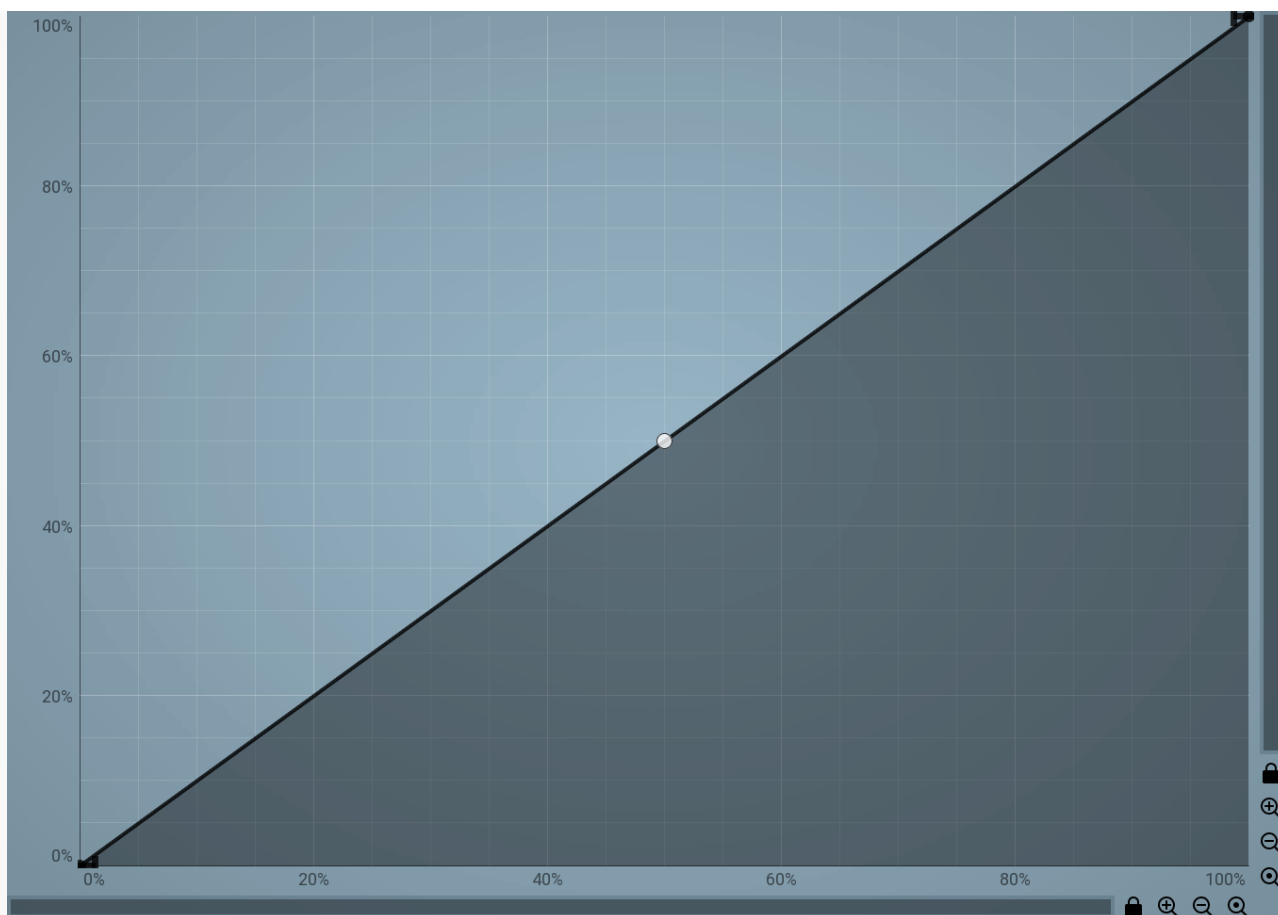
### Right arrow

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.

#### ☒ Modulation wheel is timbre

#### Modulation wheel is timbre

Modulation wheel is timbre switch makes the modulation wheel change the timbre controller as well. This is often useful for various 3D controllers, which should send timbre according to the MPE standard, but send modulation wheel instead, probably because it is more supported.

#### ☒ Timbre is bipolar

#### Timbre is bipolar

Timbre is bipolar switch makes the timbre controller bipolar meaning that per-voice modulation will produce values from -100% to +100% as opposed to other controllers, which produce 0% to 100% (hence always in the modulation direction). This is usually advantageous, since the controller itself is bipolar (meaning it's default value is the center), so it simplifies editing sounds featuring this controller.

#### ☒ Timbre defaults to minimum

#### Timbre defaults to minimum

Timbre defaults to minimum switch makes the timbre controller default value minimum as opposed to typical middle value. This gets often useful if you want the timbre to work with various controllers, some of which have the timbre, others don't, and control 'amount' of something. With the default behaviour having the timbre 50% would make the modulation running even if there is no timbre controller. Enabling this option would initially make the timbre disable the controller.

## Pitch-bend is bipolar

## Pitch-bend is bipolar

Pitch-bend is bipolar switch makes the pitch-bend controller bipolar meaning that per-voice modulation will produce values from -100% to +100% as opposed to other controllers, which produce 0% to 100% (hence always in the modulation direction). This is usually advantageous, since the controller itself is bipolar (meaning it's default value is the center), so it simplifies editing sounds featuring this controller.

## Linear smoothing

## Linear smoothing



















Linear smoothing activates the linear smoothing, which provides more accurate controller values, however since it produces equal smoothing speed for any change of the controllers, it requires higher smoothing times than the default exponential smoothing.

## Timbre uses initial-64 mode

## Timbre uses initial-64 mode

Timbre uses initial-64 mode switches on the MPE initial-64 mode for the timbre controller. Enabling this option means the timbre control is always set to value 64 at the note-on, and all subsequent changes of the timbre control relative to its actual initial value are added to that initial value of 64 to produce the actual timbre control value used by the plugin.

# Controller smoothing panel

CONTROLLER SMOOTHING				Presets		     Random 					
Pressure		10 ms		Timbre CC#74		10 ms					
Pitch-bend		10 ms		Pitch-bend for pitch		10 ms					
Modulation wheel		10 ms		Breath CC#2		10 ms					
Expression CC#11		10 ms		Sustain		10 ms					
Frequency		10 ms		Velocity off		10 ms					
Main 1		10 ms	Main 2		10 ms	Main 3		10 ms	Main 4		10 ms
Main 5		10 ms	Main 6		10 ms	Main 7		10 ms	Main 8		10 ms
Main 9		10 ms	Main 10		10 ms	Main 11		10 ms	Main 12		10 ms
Main 13		10 ms	Main 14		10 ms	Main 15		10 ms	Main 16		10 ms

Controller smoothing panel controls smoothing for various MIDI controllers. It avoids abrupt changes caused by low resolution of the MIDI standard and the MIDI controllers themselves. Smoothing is always represented as minimum time it gets for the controller to go from minimum to maximum value.

## Presets

## 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

Random button generates random settings using the existing presets.

## Pressure

## Timbre CC#74

## Pitch-bend

## Modulation wheel

## Breath CC#02

## Expression CC#11

## Sustain

## Note-off

## Tab

## selector

Tab selector switches between subsections.



## 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.



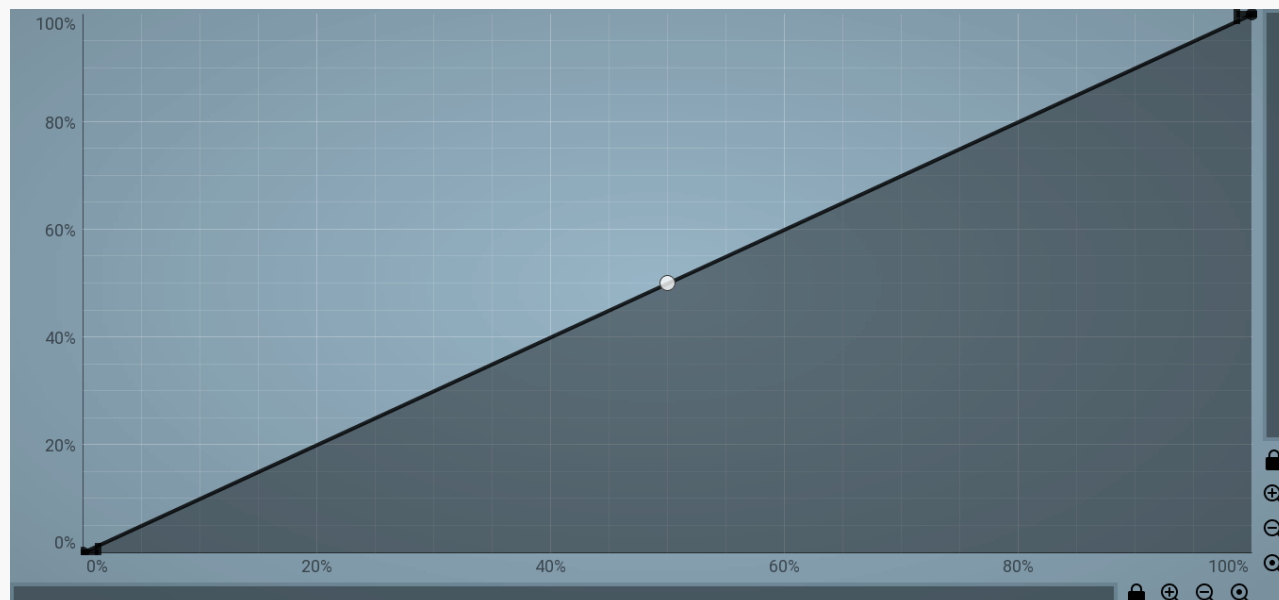
### Right arrow

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.



## 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.



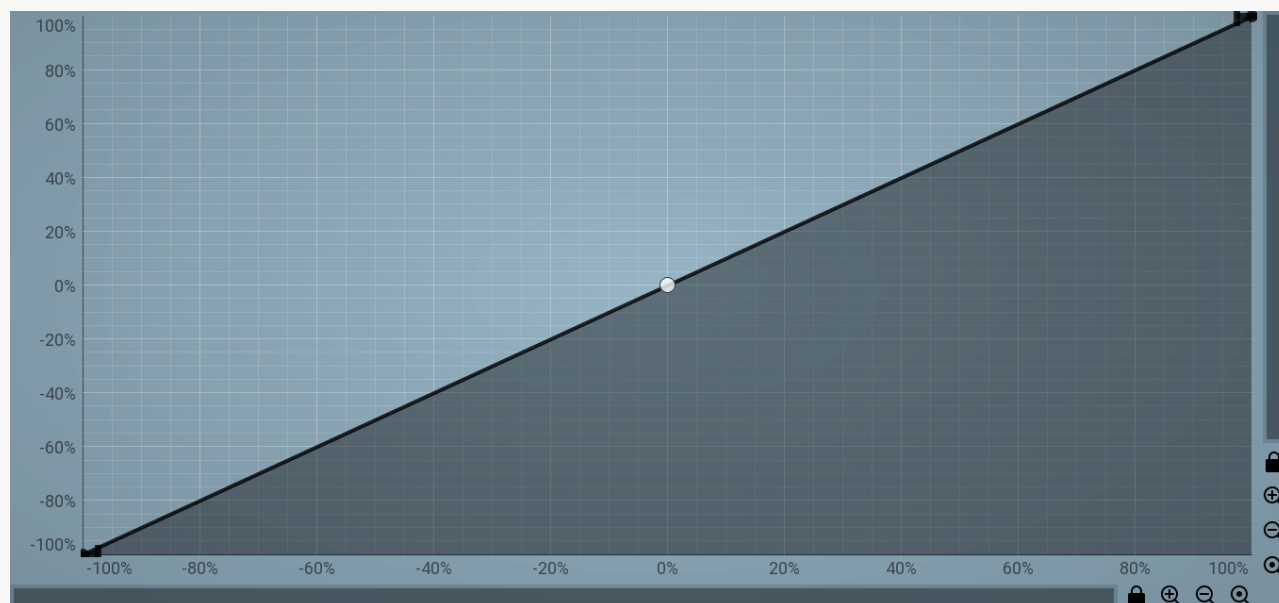
## Right arrow

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.



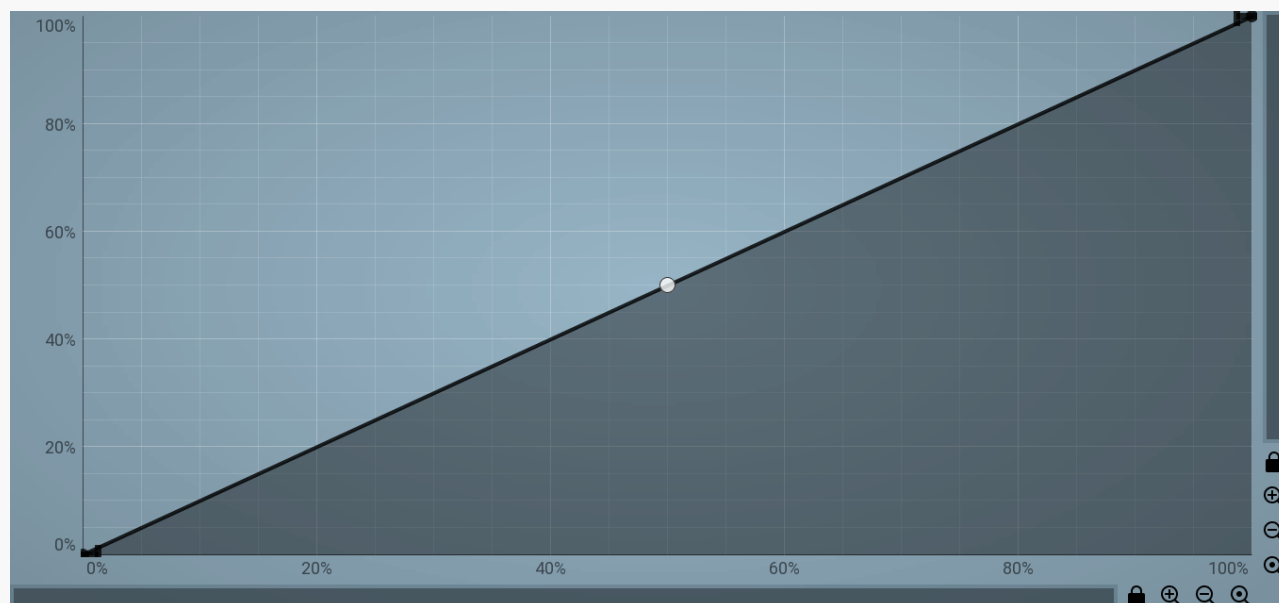
## Right arrow

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.



## 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.



### **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.





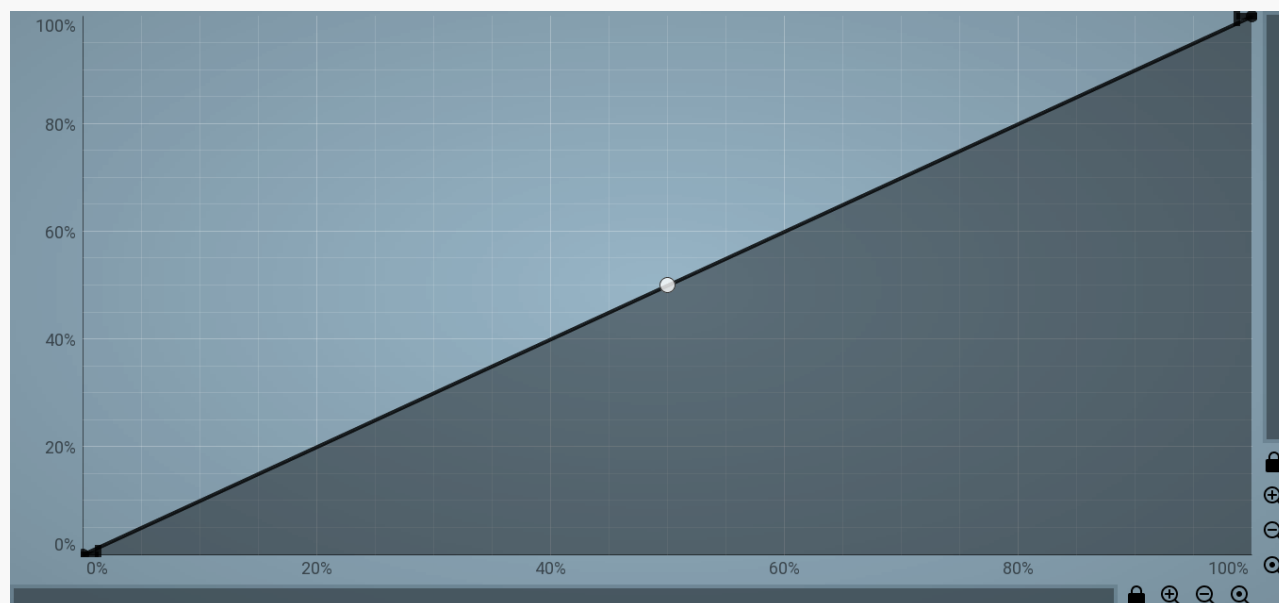
### Right arrow

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.



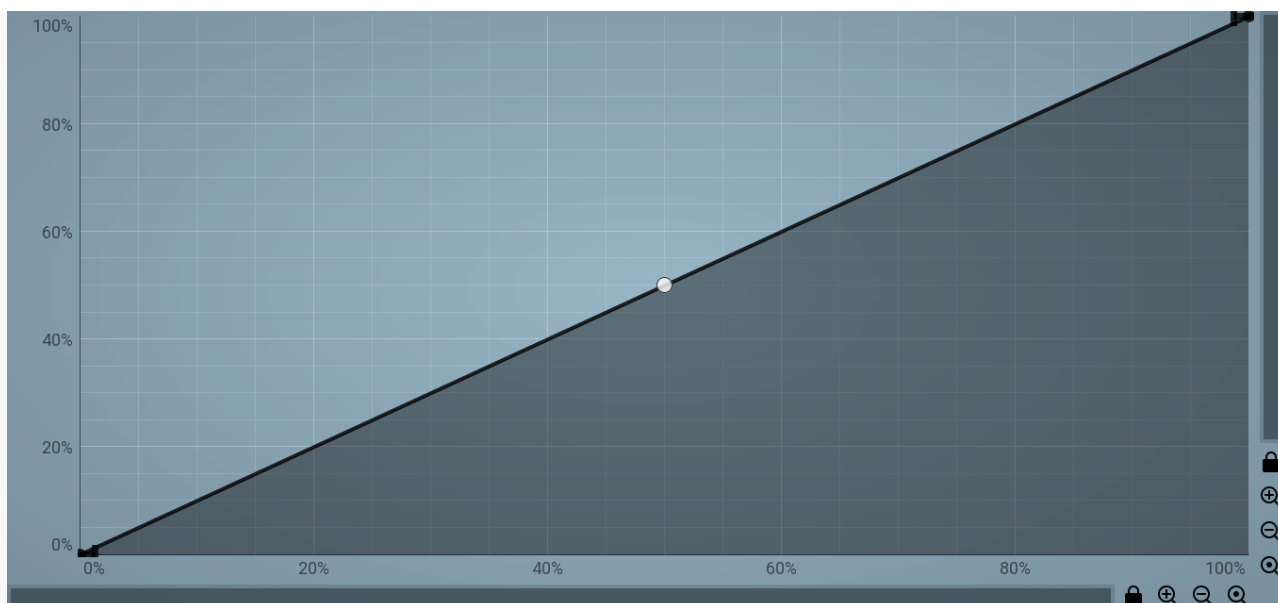
### Right arrow

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 instrument. When you press a note, the plugin may behave as if you had pressed additional notes with different velocities, tuning, delays etc.

## Presets

### 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

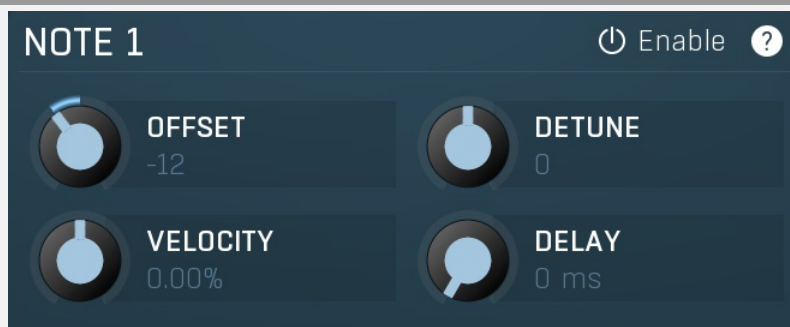
Random button generates random settings using the existing presets.



#### 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



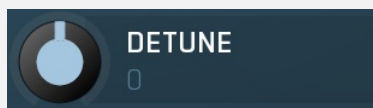
Note panel contains parameters of a virtual note. If you enable the note and press a key, the instrument will behave as if you had pressed another key at the same time, with different velocity, tuning, delay etc. Note that the keys cannot be duplicates, so if you for example hold C1 and there is a virtual 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.

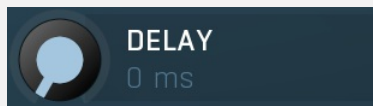
Range: -100.00 to +100.00, default 0



### Velocity

Velocity controls the difference in velocity of the virtual note from the original one.

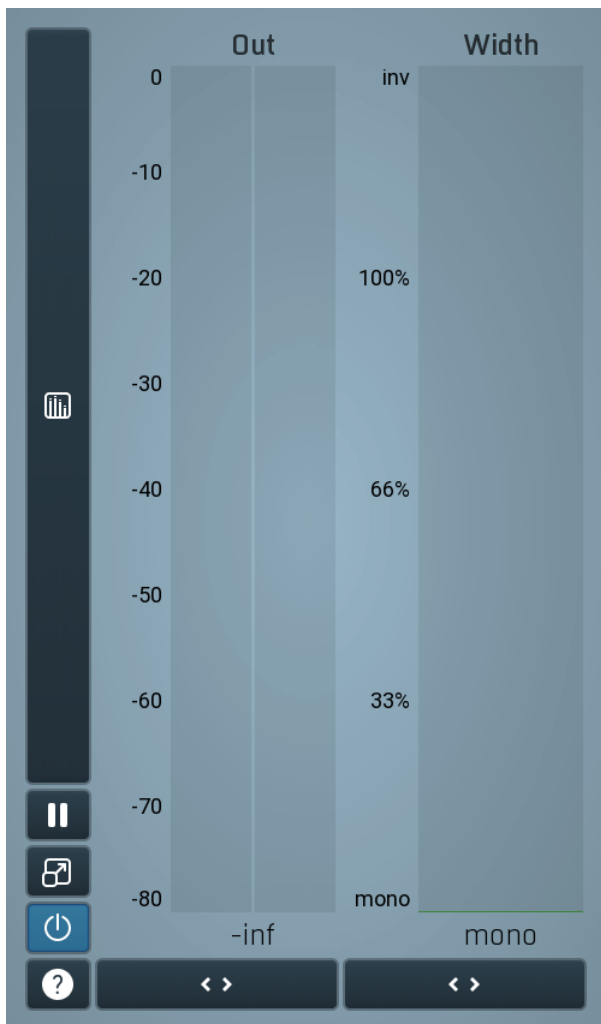
Range: -100.0% to 100.0%, default 0.00%



### 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

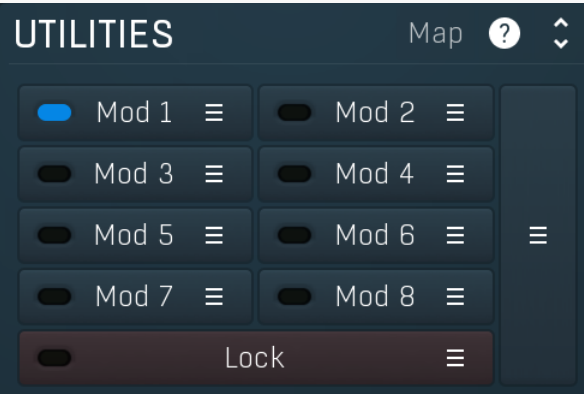
Collapse button minimizes or enlarges the panel to release space for other editors.



## 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).

## Mod 1 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.

## 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.

## 1 : Volume -54.9 dB 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.

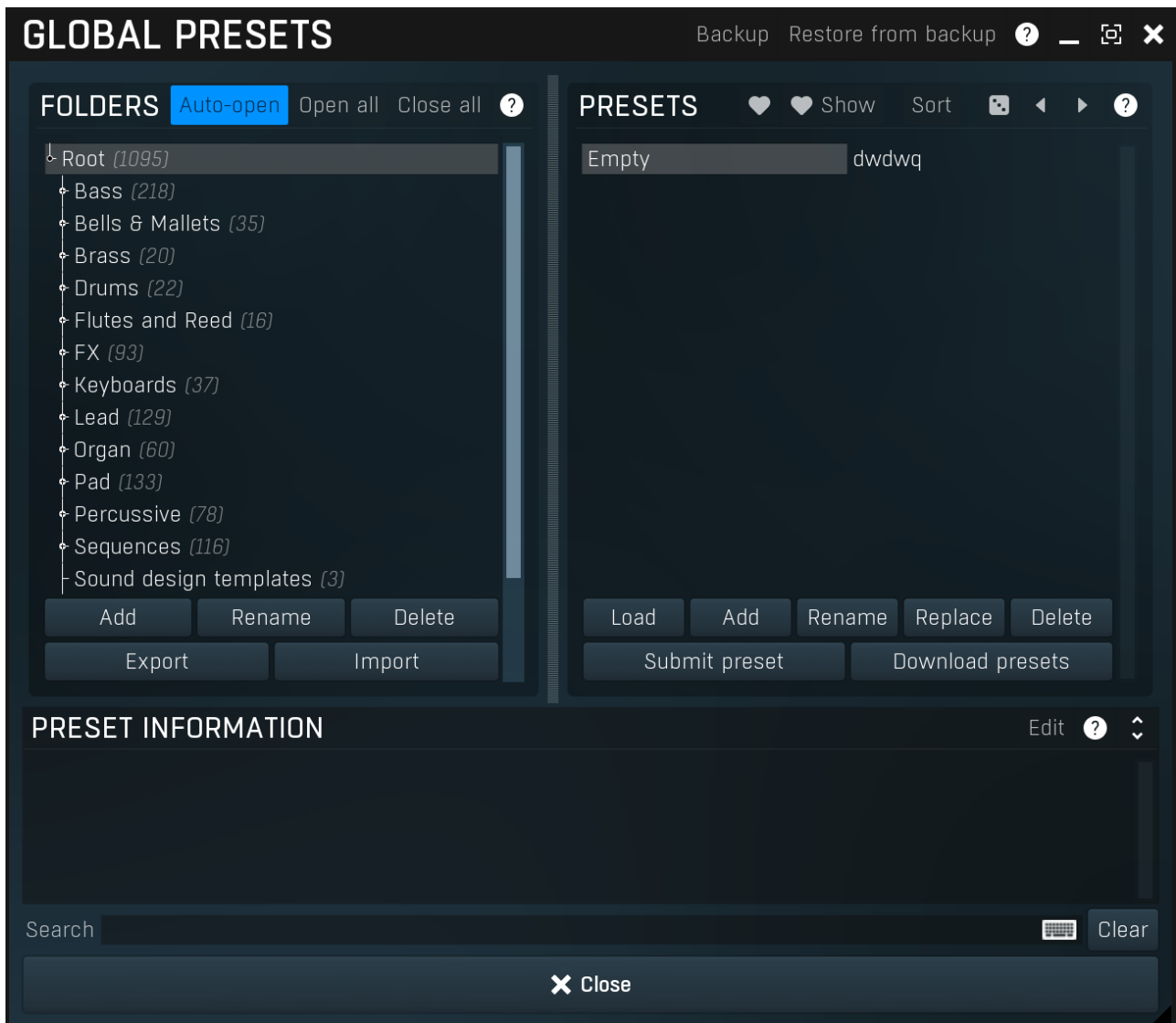
In learning mode the multiparameter does not operate but rather records your actions. You can still adjust every automatable parameter and use it normally. When you change a parameter, the plugin associates that parameter with the multiparameter and also records the range of values that you set.

*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.*



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

# Preset selector



Preset management window provides management for your presets.

Backup

## Backup

Backup button lets you backup presets for all MeldaProduction software into a single file, so you can transfer it to a different machine and restore the presets there for example.

Restore from backup

## Restore from backup

Restore from backup button lets you restore presets for all MeldaProduction software from a single file created by the **Backup** button.

## Folders tree



Folders tree lets you organize your presets into any number of folders. Use the buttons at the bottom of the window to create, rename or delete sub-folders. Note that these are not actual files & folders on disk, but are records in the preset database.

Auto-open

### Auto-open

Auto-open switch makes the tree automatically open selected items, so that all sub-folders are visible, whenever you select one. This makes it easier to browse through large structures containing many folders. The switch also makes the browser show all presets available in the selected folder including all sub-folders (except when you select the root folder).

Open all

### Open all

Open all button expands the whole tree, so you can see all of the folders. This may be handy when editing large preset structures.

Close all

### Close all

Close all button collapses the whole tree except for the root folder. This may be handy when editing large preset structures.

Add

### Add

Add button creates a new folder in the tree

Rename

### Rename

Rename button lets you rename the selected folder.

Delete

### Delete

Delete button deletes the folder including all the presets and subfolders in it.

Export

### Export

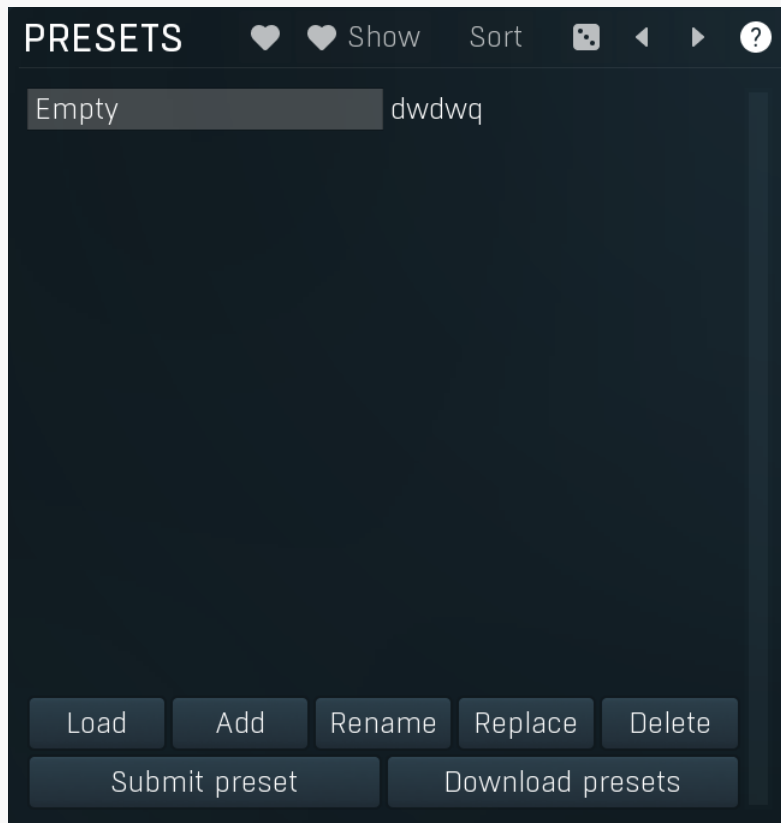
Export button lets you export the selected folder including all presets and sub-folders into a file, which you can then transfer to any computer. Or just use as a back-up.

Import

### Import

Import button lets you import a file containing presets and sub-folders and add it to the selected folder. The importer will ask you whether to destroy the original contents, so that the new presets replace previous ones, or to keep both.

# Presets list



Presets list contains all presets available in the selected folder. **Double-click** on a preset or use **Load** button to load a preset. Use the buttons at the bottom of the list to perform additional changes. Please note that these are not actual files & folders on disk, but are records in the preset database.



## Favourite

Favourite button toggles the 'favourite' indicator for the selected preset.



Show

## Show

Show button shows only the favourite presets and hides the others.



Sort

## Sort

Sort button shows the presets sorted alphabetically.



## Random

Random button selects and loads a random preset from the current folder. This way you can quickly browse the presets in the folder in a completely random order.



## Previous

Previous button selects and loads the previous preset from the current folder.



## Next

Next button selects and loads the next preset from the current folder.

Submit preset

## Submit preset

Submit preset button submits the selected preset to the online exchange servers and retrieves all the presets currently in the database. This feature serves as an online database of presets available for all the user community. Please do not submit garbage presets.

Download presets

## Download presets

Download presets button retrieves all the presets currently in the database. This feature serves as an online database of presets available for all the user community. Please consider participating by submitting your presets as well.

Load

## Load

Load button loads the specified preset. Please note that you can do the same thing by double-clicking the preset itself or pressing the Enter key.

Add

### Add

Add button creates a new preset using the current settings.

Rename

### Rename

Rename button lets you rename the selected preset.

Replace

### Replace

Replace button replaces the selected preset by one with current settings.

Delete

### Delete

Delete button deletes the selected preset.

Search



Search

Search filters the list of available presets to those containing the keywords in name or information.

Clear

### Clear

Clear button deletes all text in the search field.

## PRESET INFORMATION

Edit

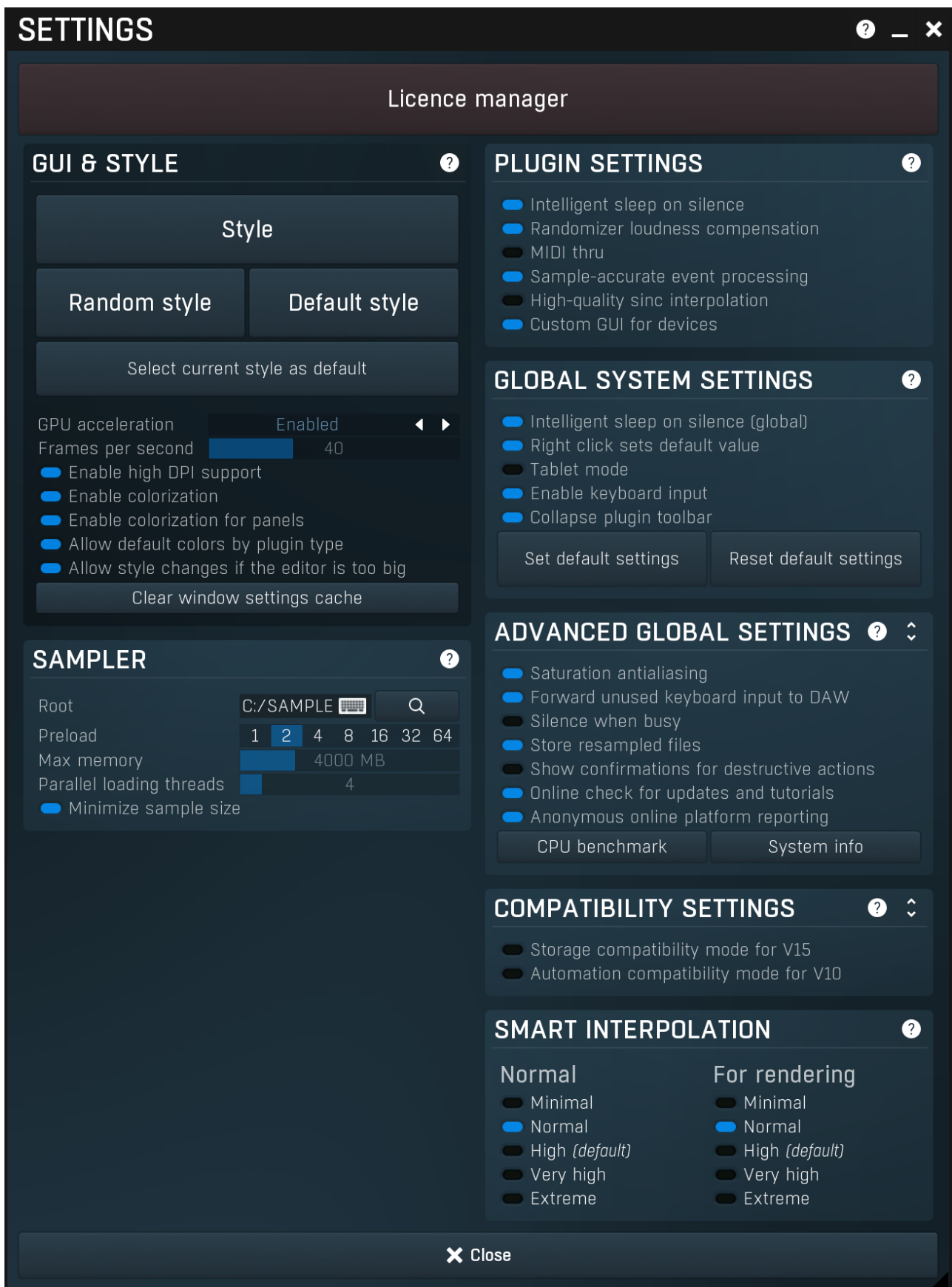


Preset

### information

Preset information field contains optional information about the preset, which you can edit when creating or renaming the preset.

# Plugin settings



Plugin settings window offers more advanced settings and is available via the Settings button.

## Licence panel

Licence manager

Licence panel lets you manage licences on this computer.

Licence manager

### Activate

Activate button lets you activate your licence for the plugin on this computer.

## GUI & Style panel



GUI & Style panel lets you configure the plugin's style (and potentially styles of other plugins) and other GUI properties.

Style

**Style**

Style button lets you change the style for this particular plugin.

Random style

**Random style**

Random style button selects a random style with random editor mode.

Default style

**Default style**

Default style button reverts to the default style and default size of the GUI. Hold the Ctrl key while clicking to revert all MeldaProduction



software products, not just the current plugin.

Select current style as default

### Select current style as default

Select current style as default button stores the current style as the default for all MeldaProduction software. This is used for the other plugins that are currently using the default style; that is, those plugins for which you have NOT selected a specific style. Please note that if you have already selected a specific style for a particular plugin, then it won't be changed until you use the Default style button.

GPU acceleration

Enabled



### GPU acceleration

GPU acceleration controls how much the GPU is used for visual rendering to save CPU power.

**Enabled mode** provides maximum speed and lets the GPU perform as many drawing operations as possible.

**Compatibility mode** uses the GPU for drawing, but doesn't use modern technologies for maximum performance. Use it if you experience occasional problems with drawing, the usual case for older ATI graphics cards. With Pro Tools on OSX this mode is always used instead of Enabled mode due to compatibility problems with this host.

**Disabled mode** disables GPU acceleration completely, drawing is then performed by the CPU. Use only if you experience technical difficulties.

A known problem may occur when using multiple displays with multiple graphical interfaces. When moving the plugin window from one display to another, it may stop displaying correctly until you move it back to the original display.

Frames per second

40

### Frames per second

Frames per second controls the refresh rate of the visual engine. The higher the number is the smoother everything is, but the more CPU it requires. You might want to lower this value if your computer is running out of CPU power.

☒ Enable high DPI support

### Enable high DPI / retina support

Enable high DPI / retina support enables the plugin to use the high resolution on high DPI (Windows) and retina (OSX) devices. It is enabled by default and detected automatically, if the host allows it. If you run into any problems, you can disable it using this option. It may be desired if you use multiple displays where only some of them feature the high resolution making the image on the low resolution ones look ugly.

If you disable this option, on Windows the high DPI device detection will be ignored and the plugin will probably appear very small. You can manually compensate for it by using a bigger style. On OSX disabling this option will disable the high DPI rendering, resulting in the classic blurry look of non-compliant applications. Changes take effect after you restart the host.

☒ Enable colorization

### Enable colorization

Enable colorization enables the plugin to change the colors of certain elements overriding your style settings. Plugins use that to highlight different parts of the graphics interface for easier workflow. You may want to disable it if you just feel it's not for you. This particular option is relevant only for controls - knobs, sliders, checkboxes etc.

☒ Enable colorization for panels

### Enable colorization for panels

Enable colorization for panels enables the plugin to change the colors of certain elements overriding your style settings. Plugins use that to highlight different parts of the graphics interface for easier workflow. You may want to disable it if you just feel it's not for you. This particular option is relevant only for containers - panels, graphs etc.

☒ Allow default colors by plugin type

### Allow default colors by plugin type

Allow default colors by plugin type is on by default and makes the plugin select its default colors depending on the type of the Plugin. Hence for instance equalizer will always be green. This is done by selecting one of the first 8 color presets for the current style, so the actual colors depend on selected style and its presets. You may want to disable this if you for example want all plugins to look the same including the style and colors. It is necessary to restart your host for a change to this option to take effect.

☒ Allow style changes if the editor is too big

### Allow style changes if the editor is too big

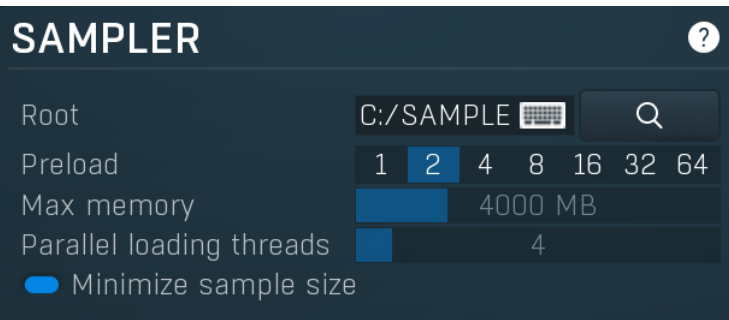
Allow style changes if the editor is too big is on by default and makes the plugin change its style, editor mode and other settings if it finds out it is too big to fit the current screen resolution.

Clear window settings cache

### Clear window settings cache

Clear window settings cache button deletes stored states of all popup windows on all MeldaProduction software. The window settings mostly contain positions and sizes, but in some cases also the data inside the popup windows. You can use this feature if something goes wrong, a window doesn't appear at all, problems like that. While this shouldn't happen and it's generally better to contract our support, this button provides a potential quick fix.

## Sampler panel



Sampler panel contains settings of the sampler module.



### Root

Root defines the main path for samples and samples banks. By placing all factory content in a single folder the plugin always knows where the data is, since these may be referenced from various places. It is recommended to use a high-speed SSD drive for this data for optimal performance.



### Preload

Preload controls how much of the samples are loaded initially when loading an instrument/project. Increasing this value makes the loading slower and increases memory consumption, but avoids possibility of required samples not being loaded in time producing audible clicks and lags. Lower this value if your computer (and specifically your hard drives) is fast enough. The compression algorithm organizes the samples typically in 64kB blocks and this value controls the number of blocks being loaded.



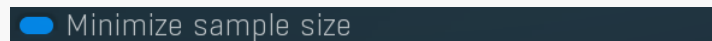
### Max memory

Max memory controls the maximum memory occupied by samples above which the engine starts releasing the unused ones. Please note that this value is only a hint, the consumed memory may exceed this value when the samples are needed. Lowering this value will reduce the memory usage, however it will also require additional CPU/disk processing to reload the samples when needed. Therefore it is best to keep this value as high as possible. If the OS runs out of system memory, it should still be able to operate properly using the so-called virtual memory, which means that unused memory blocks are swapped to HDD.



### Parallel loading threads

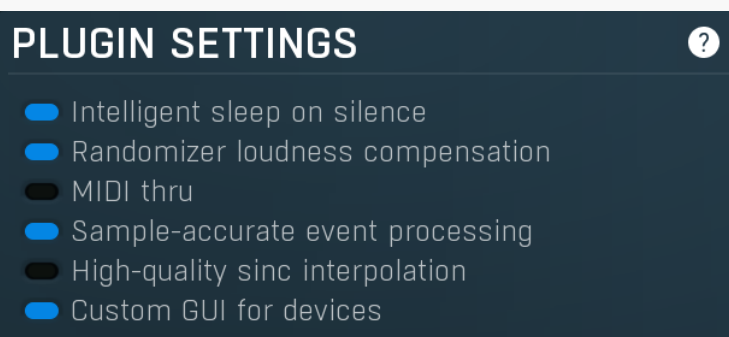
Parallel loading threads controls the number of threads that can perform loading. This option allows the sampling engine to exploit the modern fast SSDs to load big sample libraries. However if you have an old system using a hard drive, set this to 1 for optimal performance. Otherwise, the faster your SSD is, the higher you can set this value to. But please note that at some point it may start interfering with the actual audio processing. Therefore it is recommended to keep this default unless you experience glitches.



### Minimize sample size

Minimize sample size makes the sampling engine lower the memory overhead by keeping the samples in a semicompressed form. Note that this will lower the memory consumption at the expense of potentially slightly higher CPU consumption.

## Plugin settings panel



Plugin settings panel contains settings that control the behaviour of this plugin instance. These are properties that rarely need to be changed, so they have been moved here.



### Intelligent sleep on silence

Intelligent sleep on silence option provides a huge CPU saver by automatically disabling the plugin processing if the input is silent and if the plugin doesn't generate some signal on its own. This makes the plugins take virtually no CPU if there is no need for them to actually process anything. Disable this if you run into any problems with them.



### Randomizer loudness compensation

Randomizer loudness compensation enables the automatic detection of loudness after new settings have been generated using the main

**Random** button and using the output gain of the plugin to get some predefined level. This is useful in most cases since normally randomized settings can produce various output levels, so this can mitigate the problem.

#### ☒ MIDI thru

#### **MIDI thru**

MIDI thru makes the plugin pass all input MIDI through to its MIDI output. That is often advantageous in DAWs such as Reaper, which naturally pass MIDI from one plugin to the next.

#### ☒ Sample-accurate event processing

#### **Sample-accurate event processing**

Sample-accurate event processing makes the plugin schedule every event such as MIDI or automation to their accurate locations with sample accuracy, if the host allows it.

For example, if the block size in your host's audio settings is 1024 samples, this means the plugin is probably processing blocks of 1024 samples, in 44100 Hz sampling rate it is about 23ms. If this setting is disabled, any change in automation, MIDI, modulation etc. may then be granularized to 23ms (once per block), which means that you will not be able to recognize events that occur say 10ms apart from each other. When this setting is enabled however, the plugin divides processing blocks to sub-blocks and processes the events at their correct positions. This may, of course, require more CPU power.

#### ☒ High-quality sinc interpolation

#### **High-quality sinc interpolation**

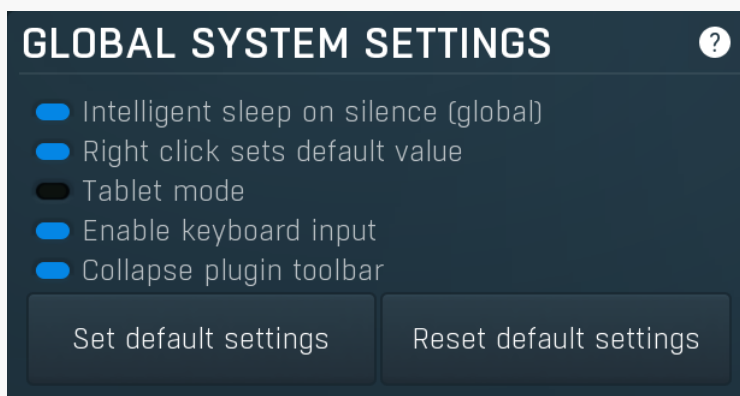
High-quality sinc interpolation activates the maximum audio quality interpolation algorithm, which is used by several modulation processors, such as a flanger or chorus. Unlike standard interpolation sinc interpolation avoids distortion and improves response for high frequencies. The disadvantage is that it introduces some latency and requires much more CPU power.

#### ☒ Custom GUI for devices

#### **Custom GUI for devices**

Custom GUI for devices enables displaying custom GUIs for the easy screen devices. You can disable it if you like the generic GUI better.

## Global system settings panel



Global system settings panel contains settings which are applied to all plugins on this computer.

#### ☒ Intelligent sleep on silence (global)

#### **Intelligent sleep on silence (global)**

Intelligent sleep on silence (global) is a global switch, which disables the **Auto disable on silence** feature in all plugins on the system. It is provided "just in case" something goes wrong.

#### ☒ Right click sets default value

#### **Right click sets default value**

Right click sets default value makes the engine set default value to a parameter when you right click on it. By default, a menu is displayed instead, with an option to set the default value, but potentially with more features. When this is disabled, you can still set a default value by holding ctrl/cmd when right clicking the control.

#### ☒ Tablet mode

#### **Tablet mode**

Tablet mode enables better support for tablets at the expense of the mouse. Enable this if you are using a tablet to control the plugins and it is behaving incorrectly.

#### ☒ Enable keyboard input

#### **Enable keyboard input**

Enable keyboard input enables the keyboard input for the main plugin window. You may want to disable if the plugin intercepts spacebar key (often used by the host for playback enable/disable and your host doesn't allow for the problem itself).

#### ☒ Collapse plugin toolbar

#### **Collapse plugin toolbar**

Collapse plugin toolbar makes all plugins collapse the plugin toolbar containing more advanced features such as channel modes, A-H presets, oversampling, safety limiter etc. It is enabled by default to make the user interfaces cleaner and easier to grasp for beginners.

Set default settings

### Set default settings

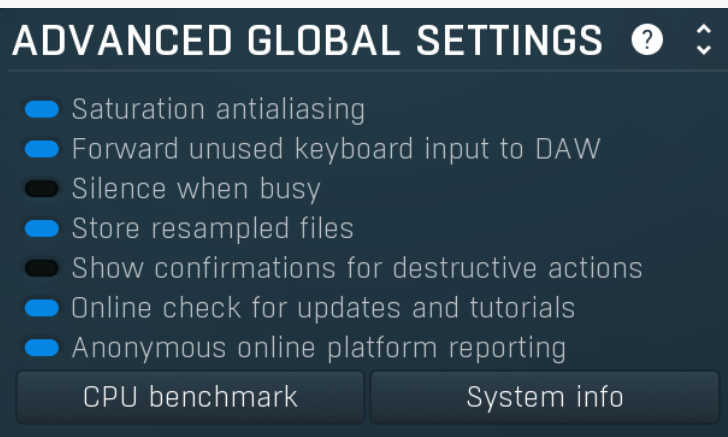
Set default settings button stores the current plugin settings as the defaults, so that when you open a new instance of the plugin, these settings will be loaded automatically.

Reset default settings

### Reset default settings

Reset default settings button removes the defaults that you set using **Set default settings** button, so that when you open a new instance of the plugin, the factory defaults will be loaded.

## Advanced global settings panel



Advanced global settings panel contains advanced settings which are applied to all plugins on this computer.

☒ Saturation antialiasing

### Saturation antialiasing

Saturation antialiasing enables a global support for antialiasing in saturation algorithms available in many of the plugins. These require additional CPU processing, however significantly reduce aliasing artifacts without a need for oversampling.

☒ Forward unused keyboard input to DAW

### Forward unused keyboard input to DAW

Forward unused keyboard input to DAW makes the plugin forward unused keyboard events to the DAW from its popups. If this is disabled, pressing say spacebar commonly used to start/stop playback won't work if a popup window is active. Enabling this makes this work and it is optional just in case your DAW does something unexpected.

☐ Silence when busy

### Silence when busy

Silence when busy makes all plugins silence the output when something time consuming is being performed in background and the plugin needs to wait for it. For instance, in modular plugins such as MXXX, adding a module requires lots of changes in the entire engine, so it is performed in background and while the plugin is inconsistent state, it is temporarily bypassed. Sometimes however, when performing live, bypassing makes the dry signal go through and that may not be wanted. So you can enable this option, and the plugin will silence the output instead.

☒ Store resampled files

### Store resampled files

Store resampled files allows the plugins create audio files for sampling rates being used if they differ from the original file sampling rate. It is used only by a few plugins, but it can improve the loading performance a lot at the cost of some additional storage on the hard drive. Disable this option if you are short on free space.

☐ Show confirmations for destructive actions

### Show confirmations for destructive actions

Show confirmations for destructive actions makes the plugin display a confirmation window whenever you are going to change the plugin settings irreversibly when using a feature, for example: when resetting your settings.

☒ Online check for updates and tutorials

### Online check for updates and tutorials

Online check for updates and tutorials lets the plugin ask about once a week if there is a new version or tutorial available so you can be easily kept up to date.

☒ Anonymous online platform reporting

### Anonymous online platform reporting

Anonymous online platform reporting helps us maximize compatibility with your operating system and host. If enabled, our plugins will send information about the system and host that you are using. We can use this information to find out which plugins and platforms are used the most and maximize testing and support there. Platform reporting is completely anonymous and requires only minimal internet connection time (a few kB once a week).

## CPU benchmark

### CPU benchmark

CPU benchmark button calculates the performance of the plugin with the current settings.

## System info

### System info

System info button displays some technical information about the build and the machine.

## Compatibility settings panel

### COMPATIBILITY SETTINGS

- ☐ Storage compatibility mode for V15
- ☐ Automation compatibility mode for V10

Compatibility settings panel contains advanced settings you rarely need unless you run into some problems when using multiple versions or old projects.

#### ☐ Storage compatibility mode for V15

#### Storage compatibility mode for V15

Storage compatibility mode for V15 reverts to the older and much slower storage system used by version 15 and older. Use this if you want to open your projects or presets on older version of MeldaProduction plugins.

#### ☐ Automation compatibility mode for V10

#### Automation compatibility mode for V10

Automation compatibility mode for V10 reverts the set of automation parameters back to version 10 and earlier. Use this if you need the plugins to work with projects, which contain automation, made using version 10 or older. In version 11 the list of automatable parameters have been highly simplified and reorganized and multiparameters are provided for the vast number of hidden parameters. This should speed up loading, improve workflow with the plugins and improve compatibility with various hosts.

## Smart interpolation panel

### SMART INTERPOLATION

#### Normal

- ☐ Minimal
- ☒ Normal
- ☐ High *(default)*
- ☐ Very high
- ☐ Extreme

#### For rendering

- ☐ Minimal
- ☒ Normal
- ☐ High *(default)*
- ☐ Very high
- ☐ Extreme

Smart interpolation panel controls the depth of the smart interpolation algorithm, which controls the parameters in order to provide maximum audio quality and lower the chance of zipper noise. Smart interpolation is engaged whenever you change any parameter via the GUI, modulators, multiparameters, MIDI or automation.

Many parameters can be automated easily and the plugin responds with sample-accurate results. However, several parameters need exhaustive pre-processing when changed. In these cases, the parameters are not updated every sample, but, for example, once every 32 samples. This highly reduces CPU usage, but affects the output quality.

With modulators the situation is more complicated. Besides the updating issue, the modulator itself can perform some pretty advanced processing, hence it is better to perform the processing in blocks. However, the bigger the block, the less often the modulator updates those parameters associated with it and the resulting modulation is less accurate. In a way you can say that the modulator is slower and lazier. This may actually be wanted, so when it comes to modulators it is not true that a better mode always means better output quality.

The smart interpolation mode controls the maximum number of samples being processed before the parameters are updated. **Minimal mode** uses 2048 samples and rarely will do anything unless processing offline. **Normal mode** uses 256 samples and usually is enough to achieve good quality results. **High mode** uses 32 samples and provides perfect quality for most cases. It is also a good compromise

between CPU usage and audio quality, so it is the default. **Very high mode** uses 4 samples and you will rarely need it. **Extreme mode** uses 1 sample, which means that everything is updated after every single sample. This provides the highest possible accuracy and quality you can ever achieve, however it requires lots of CPU and it is very unlikely that you will ever need it. If you use this mode and still hear audio artifacts, then either what you are hearing is actually CPU overload, or you are doing something that is not physically possible.

The higher the mode, the quicker the parameter updates, but the more the CPU load.

*Please note that modulating certain parameters without artifacts is impossible. For example, when modulating a delay very quickly, the physics of such a process just cannot occur in the natural world and the results are appropriately unnatural. These physically impossible processes usually manifest themselves as distortion or zipper noise.*

# Modulator editor



Modulator is an extremely advanced feature, which lets you change parameters automatically depending on various inputs. You can use this to add movement to your sound, respond to some plugins differently for louder sections, or even follow the pitch of the input.

The modulator edit window has two parts: on the left side you can configure the mode of the modulator (the way the modulator works) and on the right side there is a list of parameters to modulate. A modulator can control all automatable parameters (and often more than that) including the parameters of other modulators. Each modulator can control as many parameters as is needed and each of the parameters has its own range and transformation shape. The values and ranges of the first 4 parameters associated with the other modulators can also be modulated/automated. The following modulator modes are available:

**Normal** mode makes the modulator behave like an ordinary low-frequency oscillator (LFO). There are various ways to control its shape as with all oscillators in our plugins. Each modulator can synchronize to the host in the **Synchronization** panel. Modulators can also synchronize with each other using the **Sync groups**. Using **MIDI reset** you can reset the oscillator to any phase using MIDI notes, but obviously to-host synchronization must be disabled in order for this to work.

Note that the settings in this mode are used even if the modulator is actually in a different mode by using "LFO modulation". This basically blends between the actual mode, which may for example detect the input signal level, and give it some additional movement using the LFO depending on the LFO modulation parameter available for each of the remaining modes.

**Follower** mode makes the modulator detect the input signal level. It contains an extremely advanced and accurate level detector taken from our MDynamics plugin. The level follower is an immensely useful feature, yet it may be a little difficult for beginners to comprehend, so we will cover it here in more detail.

It is often necessary to adjust the follower slightly for new material. First, it has the standard parameters - attack, release, hold and RMS length. These are fairly standard features and help is available for each of them. **Level min and max** controls the range of input levels. When the input level is equal to or below the min level, the modulated parameters' values will be minimal. Similarly, when it reaches the max level, the modulated parameters' values will be at their maximum. This allows for adjustments to the range of input levels, which are certainly different for any audio material and settings. It can be used creatively too - for example, by using very low values for both limits we can differentiate between silent and non-silent parts, similar to the way a gate effect works.

**Advanced detector settings** provide some extraordinary features, such as psycho-acoustic pre-filtering, which forces the modulator to detect loudness instead of raw input levels, custom input signal pre-filtering using a fully featured 6-band equalizer, and custom attack and release shapes. **Band-pass panel** pre-filters the level detection signal using a band-pass filter, so this is like a very simplified version of the equalizer from the advanced detector settings. **Side-chain** makes the modulator measure side-chain input if the plugin has one. For modular plugins the modulator can also be driven by a feedback signal. The **advanced panel** provides some further level processing features that you can take advantage of creatively or to further adjust to your actual audio material.

**Project onto LFO shape** is a more advanced concept, which is available for other modulator modes too. You can easily imagine, that the



modulator in any mode generates values for each parameter, we can say it is between 0 and 1, where 0 sets minimum parameter value, and 1 sets the maximum. Project onto LFO shape forces the modulator to use this range in the oscillator shape, which can then be configured in normal mode. The value is basically transformed by the oscillator shape, where the values generated by the modulator are on the horizontal axis (phase) and the output is the actual oscillator value. This feature has no physical meaning and can only be used creatively - to transform the more or less linear results of the level follower into a much more complicated curve.

*Let us demonstrate the follower mode with an example - the idea is to apply a delay to a snare drum within a previously mixed drumset. This is commonly used on reggae/dub rhythms for example, however in these cases the snare track is usually available separately. Using the modulators you can get somewhat interesting results even with an already mixed drumset. The idea is to increase the input gain whenever the snare is playing, so that only the snare drum (and potentially other instruments playing at the same moment) are passed into the delay. So first teach the modulator to control input gain parameter of the delay and set it to follower mode, potentially configure some of the parameters to get the desired response. Now the louder the input is, the more delay you get. To make it respond only to snare drum, enable the band-pass and set the filter limits accordingly, e.g. 500Hz to 1k. This makes the input gain increased depending on the input level in this part of the spectrum, which contains the snare drum.*

**Envelope** mode causes the modulator to generate an arbitrary envelope, similar to those from synthesizers. It can either follow MIDI - the envelope starts when a key is pressed, goes through the attack and decay stages, then holds in sustain stage until the key is released when the release stage begins, or it can follow audio - when the audio level exceeds **Threshold on** it behaves the same way as when a note is pressed in MIDI mode, and then when the input level drops below **Threshold off** it behaves like a key release. As with most modes there is LFO modulation and LFO projection and the input level can be driven by the side-chain or feedback if available. The envelope shape can be adjusted using several controls (lengths of each stage etc.) and you can even draw your own shape.

**Random** mode is a smooth random generator. It is very handy if you want some parameters to change over time, but do not actually want them to be periodic like LFOs. A modulator in random mode does not actually generate random values, the results will always be the same at each position in your arrangement in the host. This allows a pseudo synchronization with the host and ensures a "what you hear is what you get" performance. **Speed** parameter controls the speed of change and any slight change to this parameter will change the whole stream.

**Pitch** detects the pitch of the input signal assuming it is not polyphonic (here it can work too and will probably detect the lowest note, however it is definitely not suitable for percussive signals, which do not have a pitch). It is very useful, enabling you to tune an oscillator to follow your singing, or allow an equalizer to control separate harmonics of a vocal, use a distortion to get more drive for higher notes in a guitar solo and much more. The pitch detection may be a little tricky to understand, so we will discuss it in more detail.

A pitch detector takes the input signal and tries to approximate the pitch of the fundamental frequency in it. It is physically impossible to detect pitch instantly, as an extreme example, 20Hz takes 50ms for the signal to evolve enough to detect that there is actually a 20Hz frequency in the signal. For this and many other reasons any pitch detector employs several limitations. These are available in the **Detector panel**. The defaults will work well for most audio material, however, it is useful to understand the parameters, so that you can let the detector adapt better to your particular audio materials if necessary, and also in order to be more creative.

**Min and max frequency** parameters in the Detector panel control the limits of the frequencies you expect in the input. For example, a female voice is unlikely to sing below 100Hz, so it is customary to set the minimum frequency to 100Hz or even higher. Voice signals contain several artifacts, blows and pops, all of which can temporarily create frequencies below the actual pitch of the voice, so setting these limits is preferable to avoid "jumps" to incorrect pitches. **Stabilization** and **Speed** also prevent these jumps by restricting how quickly the pitch can change. These can also be used creatively. **Threshold** controls the minimum level of the input signal to be considered "not-silent and probably having pitch". This acts as a form of gate, which prevents the detector from analyzing irrelevant rumble in between actual performances. **Shift panel** allows the detected pitch to be shifted up or down and **Auto-tune panel** moves it to the closest note - similar to the automatic pitch changing function from MAutoPitch, except no pitch shifting is actually done and the results are used purely to control some parameters.

**Min and max frequency** parameters in the top of the editor have a very different meaning than the parameters of the same name in the detector panel. From now on we will assume that the pitch has been detected successfully and are now considering what to do with the results. Again, we may assume the modulator generates values from 0 to 1, where at 0 the modulated parameters' values become minimal and reach maximum at 1. When the input pitch is equal or below the min frequency parameter, the modulator's value is 0, hence modulated parameters will have a minimal value as well. Similarly when the pitch reaches max frequency, the modulated parameters will get to the maximum.

Now you may say this makes no sense, because the detected pitch cannot exceed the limits specified in the **Detector panel** anyway. The reason for this is that most "frequency" parameters of all plugins are limited from 20Hz to 20kHz, whether it is the frequency of a band in an equalizer, or a high-pass frequency in a phaser for example. It is a reasonable solution since physiologically speaking these figures are on or around the range of our hearing limits.

*Let us explain the concept with an example. We want to modulate a band of an equalizer, so that it always follows the fundamental frequency, the pitch, of our audio material. All we need to do is to switch the modulator to pitch mode, allow it to control the band frequency parameter and set the range for this parameter to the full range, from 20Hz to 20kHz. The pitch detector may then detect frequencies from 50Hz to 2kHz, but the modulator takes it that the actual limits (converted to 0..1) are 20Hz to 20kHz and that exactly the same range is configured for the band frequency parameter, so you could say that "they understand each other". We did not need to touch the min and max frequency parameters at all.*

*Here is one more example, where we would actually want to adjust the min and max frequency parameters. We want to control a drive parameter of a distortion for a guitar so that the higher the guitarist plays the more distortion he gets. Again, we teach a modulator to control the drive parameter, for any range we want, and switch the modulator to pitch mode. Now the modulator will move the drive parameter, but only slightly, because it assumes the pitch can vary from 20Hz to 20kHz, but the guitar may actually only play from about 100Hz to 1kHz. So we can use the min and max frequency parameters to say "what is high and what is low", to limit the frequency range. There are no general rules here, you have to experiment, because every instrument and parameter is different.*

To sum things up, the difference between controlling a frequency parameter and a drive parameter is simply the fact that a frequency

parameter is compatible with the pitch. After all, pitch is nothing more than a frequency (strictly speaking it is a logarithmic representation of frequency).

## Presets

### 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

Random button generates random settings. Note that unlike copy & paste, presets & randomization do NOT affect the set of parameters being modified, hence it serves to optimize adjustment of the modulator behaviour assuming that you already specified the set of parameters to control.

If you hold **Shift**, the plugin will undo previous randomization.



#### R

R button enables automation read. This way you can actually automate the modulation value. First you use **W button** to record the modulator values over time. After that you can modify it in some way and enable automation read to override the normal modulator behaviour. Note that the results may be different when automation is used with potentially lower audio quality and slower response.



#### W

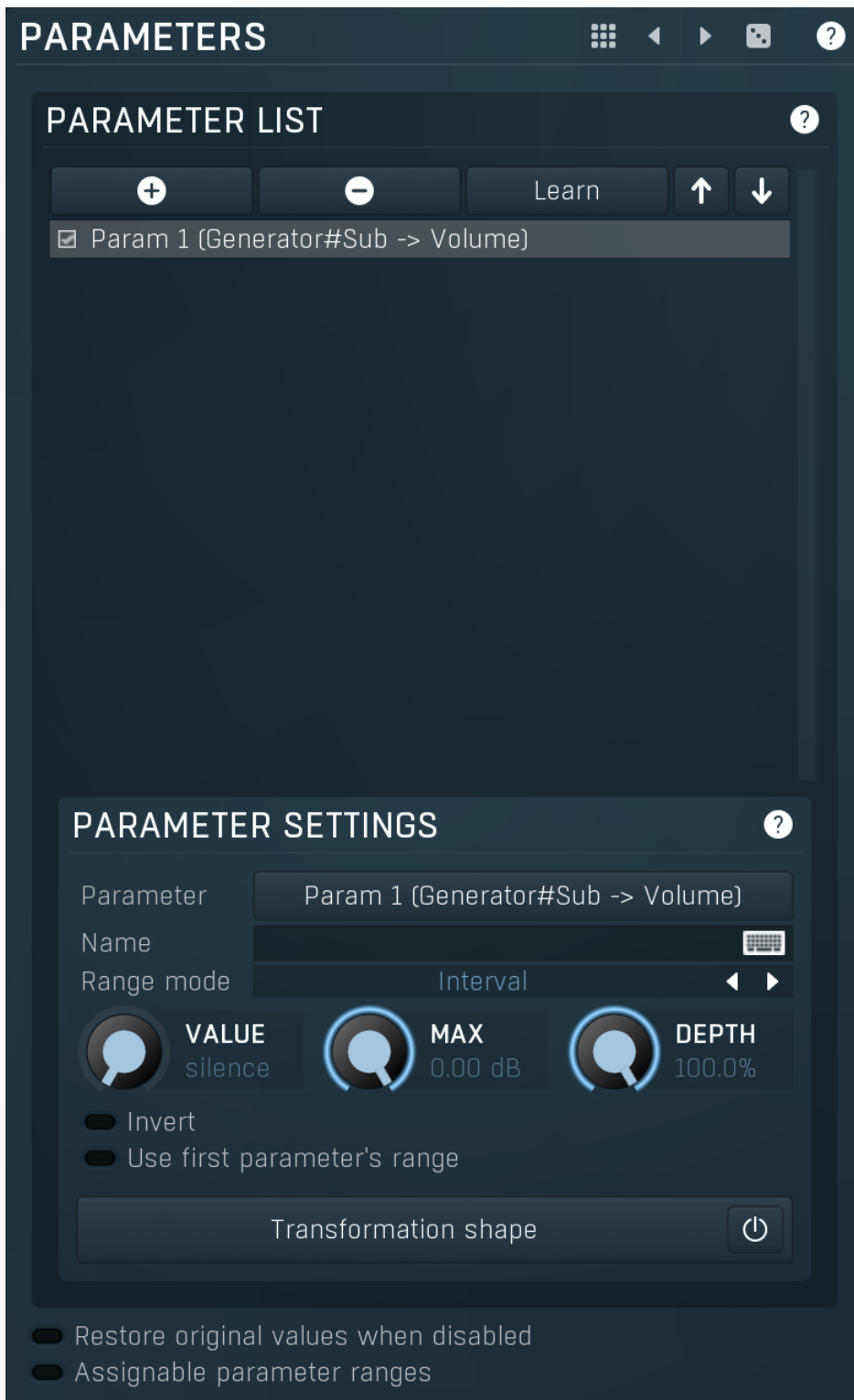
W button enables automation write. This way you can actually automate the modulation value. Use the button to record the modulator values over time. After that you can modify it in some way and enable automation read to override the normal modulator behaviour. Note that the results may be different when automation is used with potentially lower audio quality and slower response.



#### Map

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

## Parameters panel



Parameters panel contains the list of the parameters that the modulator is controlling, their ranges 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

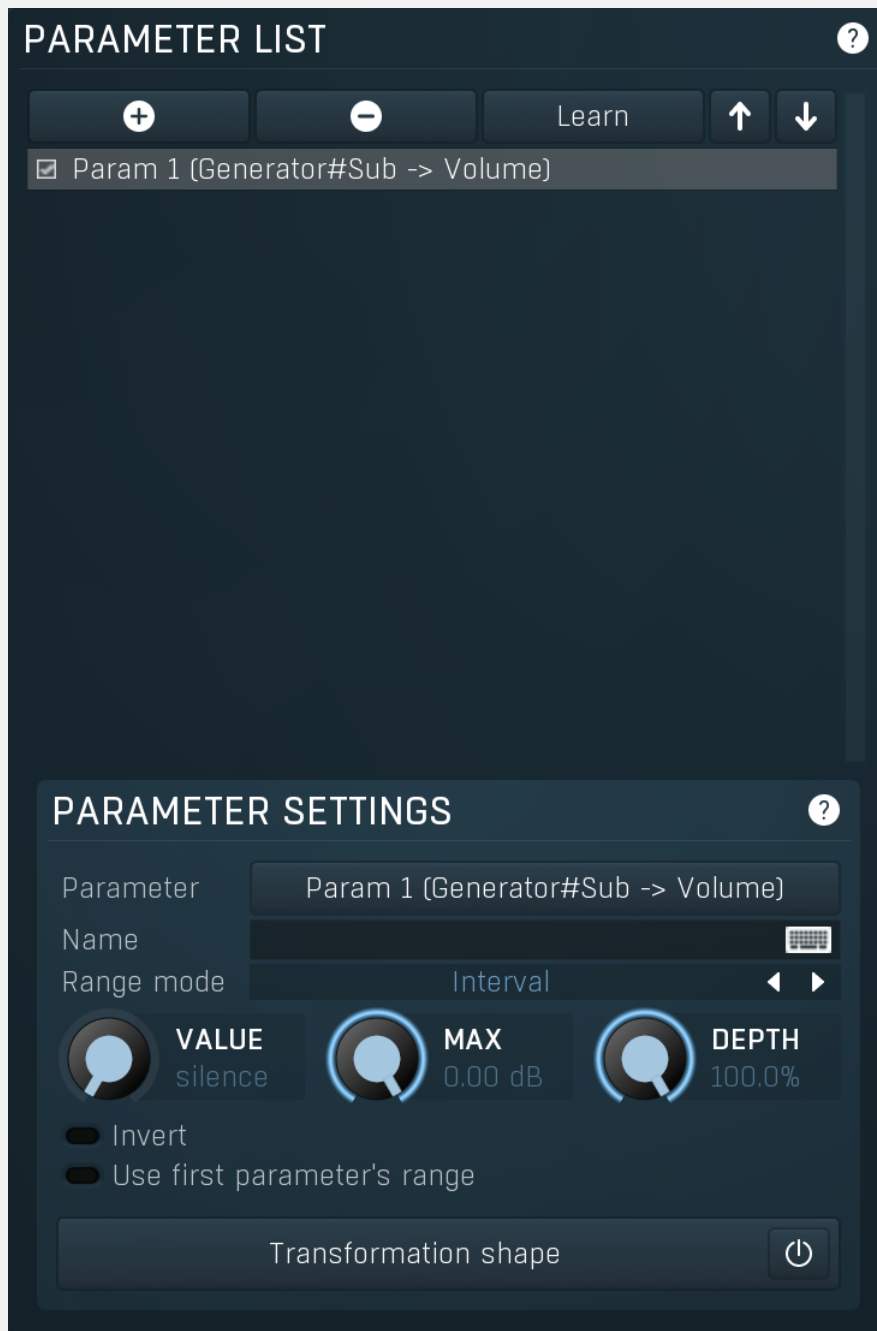
Right arrow button loads the next preset.



### Randomize

Randomize button loads a random preset.

## Parameter list



### Add

Add button adds a parameter to the list of controlled parameters. Alternatively you can use the learn feature available by right-clicking the modulator button.



### Delete

Delete button deletes the selected parameter from the list of controlled parameters.

Learn

### Learn

Learn button starts or stops the learning. Click it, then move some parameters in the plugin, then click it again. Learning can also be accessed from the global modulator menu.



### Up

Up button moves the selected parameter up one item, if possible. This may be useful when keeping things organized, but please note that if you have some other multiparameter, modulator or another subsystem access the ranges of individual parameters, this function will reorder them, so these connections will no longer be correct.

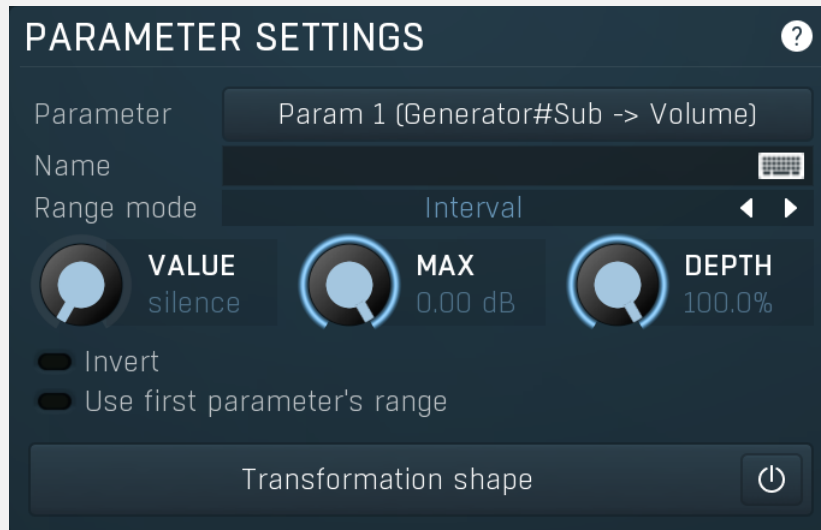


### Down

Down button moves the selected parameter down one item, if possible. This may be useful when keeping things organized, but

please note that if you have some other multiparameter, modulator or another subsystem access the ranges of individual parameters, this function will reorder them, so these connections will no longer be correct.

## Parameter Settings



Parameter Param 1 (Generator#Sub -> Volume)

### Parameter

Parameter defines the target parameter which is being modulated. The set contains all automatable parameters.

Name

### Name

Name lets you name the parameter somehow and may be helpful in situations, where there are many parameters being edited without obvious meanings.

Range mode Interval

### Range mode

Range mode defines how the parameter range is selected. While sometimes it is better to specify minimum and maximum, other times it is better to use a nominal center and depth (% of full scale). This control allows you to define which one it will be.

**Up and down** mode makes the values go above and below the selected **Value**, which is considered the center. The interval is made smaller if necessary.

**Full range mode** is similar, except the range is symmetrically constrained, so the selected **Value** may not be the center anymore.

**Up/down only modes** goes from the selected value up/down only.

Let's compare these 4 modes. Taking a value of -12dB value, with a depth of 75% and a scale of +/- 24dB. The nominal range is therefore = +/-24 dB \* 75% = 36dB. With values of 0%, 50% and 100% the outputs are:

Up and down: -24, -12, 0 (range constrained to 12 dB either side)

Full range: -24, -6, 12 (range limited to minimum, but not constrained)

Up only: -12, 6, 24 (range not constrained = +/-24 dB \* 75% = 36dB)

Down only: -12, -18, -24 (range limited to minimum)

**Interval mode** is the most simple one and goes from **Value** to **Maximal value**.

VALUE  
silence

### Value

Value defines the center of the target parameter's range or the minimum if the **Range mode** is set to **Interval**.

MAX  
0.00 dB

### Maximal value

Maximal value defines the upper limit of the target parameter's range. It is available only if the **Range mode** is set to **Interval**. This value can be lower than **Value**. 0% is always mapped to reference>Value and 100% to reference>Maximal value.

DEPTH  
100.0%

### Depth

Depth defines size of the target parameter's range. It is used only if the **Range mode** is not set to **Interval**.

☐ Invert

### Invert

Invert checkbox inverts the target parameter's range, so that minimum becomes maximum and vice versa.

### Use first parameter's range

Use first parameter's range makes the parameter display use the same range as the first parameter in the list. This is often useful if you want to control the range in some way and apply the range to multiple parameters.

### Use first parameter's range

### Transformation shape



### Transformation shape

Transformation shape button displays the graph editor, which lets you tweak the shape of the curve used to control the selected parameter. The X axis shows the original values, the Y axis defines the results. Note that this takes some CPU, therefore you have to enable it using the enable button in the title.

### Restore original values when disabled

### Restore original values when

#### disabled

Restore original values when disabled makes the modulator restore the original parameter values when it is disabled by automation or modulation. Normally when you manually disable the modulator, the original values are restored as that is usually desired. However when you control the modulator enable state by automation or modulation, you may or may not want this to happen.

### Assignable parameter ranges

### Assignable parameter ranges

Assignable parameter ranges allows you to assign parameter ranges of several first parameters to other subsystems such as multiparameters or modulators. By default it is disabled, which removes all the relevant parameters to save valuable resources. This feature is available only if automation compatibility mode for V10 is disabled.

## NORMAL

## ENVELOPE

## RANDOM

### Mode

Mode defines the way in which the modulator works. The modulator is like a black box that generates one number in range 0% to 100% at each moment and then assigns the appropriate value to each of the target parameters. The mode defines what this number will be. Select the particular tab to control the modulator's behaviour.

**Normal** mode uses a standard low-frequency oscillator (LFO) to drive the parameters.

**Follower** mode uses the level of the input signal.

**Envelope** generates an envelope using MIDI notes or by following input signal level.

**Random** generates randomized output which is however the same every time you render the song.

**Pitch** detects and follows the pitch of the input signal.

# Normal mode

MODULATOR SHAPE

Presets

◀ ▶

🎲

📄 📄

Random

?

Normal

↓

Harmonics

Shape

Sine

Custom

25.0%

Step

25.0%

Smooth

50.0%

Advanced

📐

📐

📐

📐

Edit

Edit

📐

📐

Assignable advanced shape parameters

RATE

Sync

?

Frequency

1.000 Hz

Sync group

Disabled

1

2

3

4

Reset on ARP

Length

1 / 4

◀ ▶

Type

Straight

◀ ▶

Set frequency

Phase

90° [25.0%]

Count

1

MIDI RESET - (RE)TRIGGER

🔌 Enable

?

⬆ ⬆

🔵 Note-on

🔴 Note-on only first

🔴 Note-off

🔴 Note-off only last

🔴 Single shot

🔵 Single shot reset

Min velocity

0.00%

Max velocity

100.0%

Min note

0 (C-1)

Max note

127 (G9)

Phase

0° [0%]

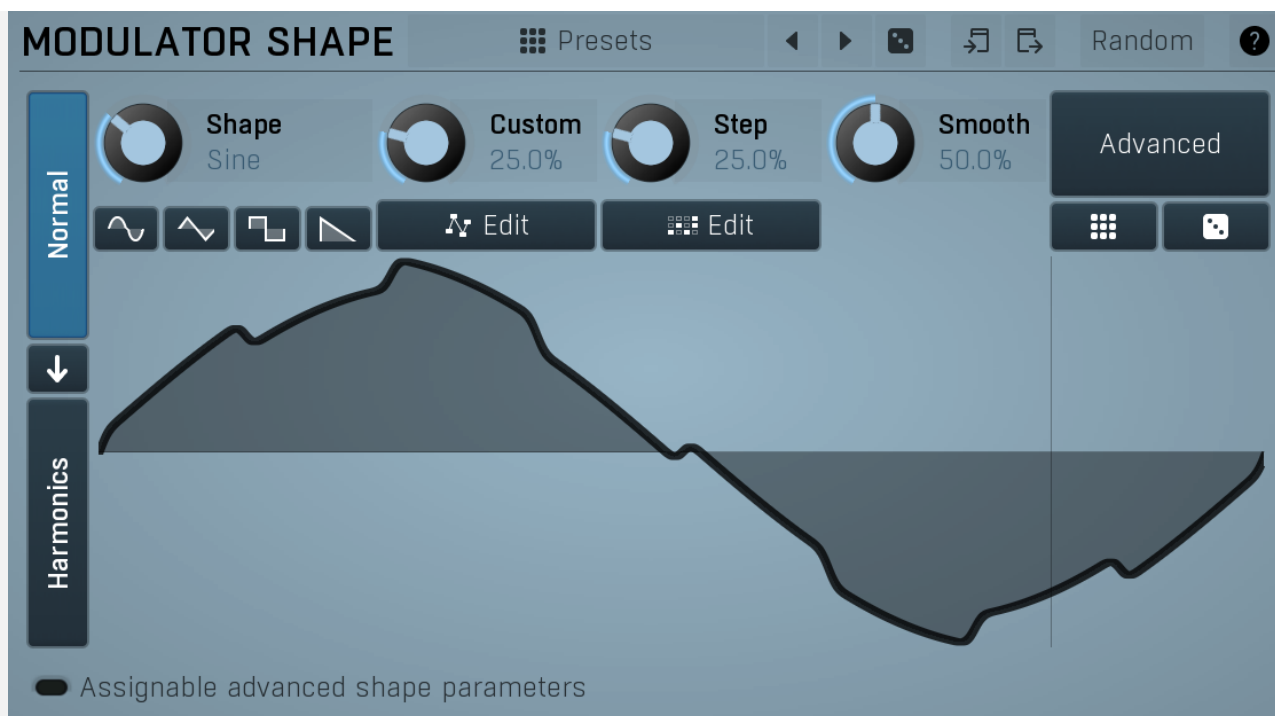
Channel

All

◀ ▶

Normal mode makes the modulator work as a traditional low-frequency oscillator (LFO). Note that even if the modulator itself is running in a different mode, you can still blend this LFO using the **LFO modulation** parameter available on each tabbed page. The LFO parameters themselves are available on the first tabbed page only though.

## Signal generator



Signal generator defines the modulation LFO shape. It is used by the LFO generator, but also for the **Project** feature. 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?  $50 \text{ Hz} * 256 = 12.8\text{kHz}$ . 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 ( $3 \times 220$  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

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

Random button generates random settings using the existing presets.

Normal

#### 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

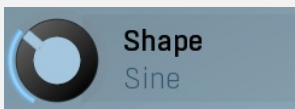
#### Harmonics

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

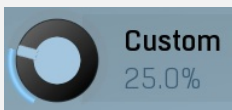


Signal generator in Normal mode works by generating the oscillator shape using a combination of several curves - a predefined set of standard curves, custom shape, step sequencer and custom sample. It also post-processes the shape using several filters including smoothing to custom transformations. This 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.



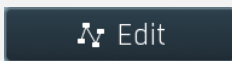
### Shape

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.



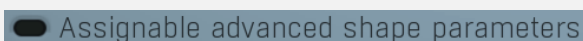
### Smooth

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

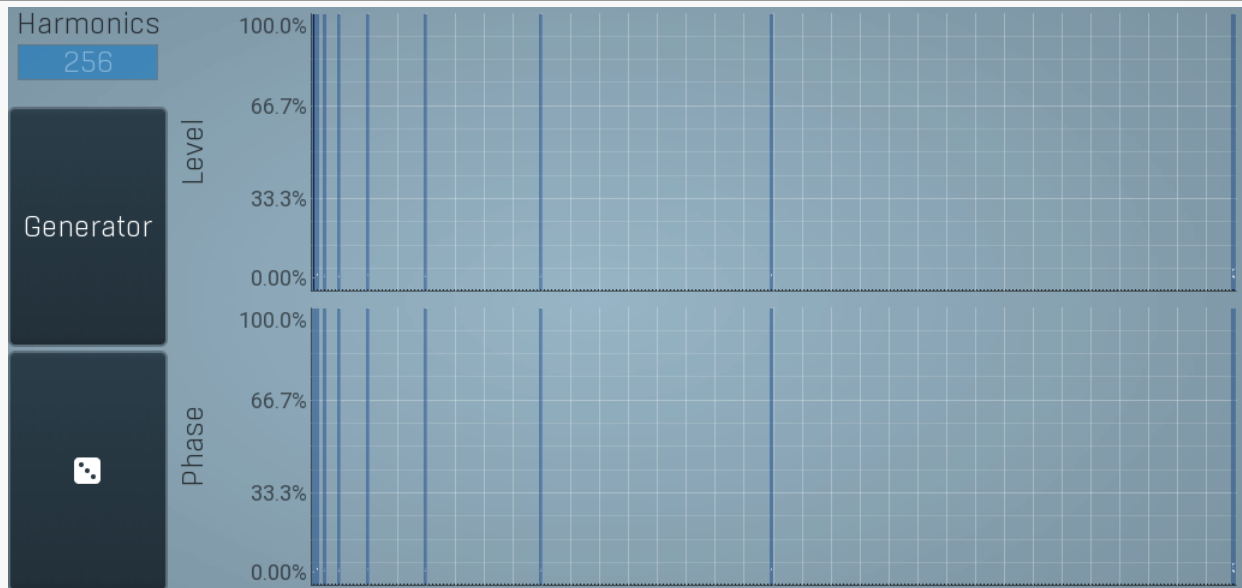
Advanced button displays an additional window with more advanced settings for post-processing the signal shape, such as harmonics or custom transformations.



### Assignable advanced shape parameters

Assignable advanced shape parameters allows you to assign advanced parameters such as step sequencer values to other subsystems such as multiparameters or modulators. By default it is disabled, which removes all the relevant parameters to save valuable resources.

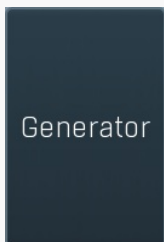
## Signal generator in Harmonics mode



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.



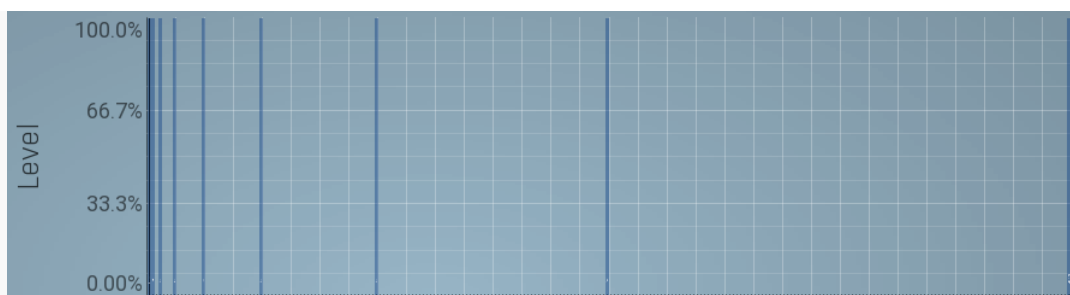
**Generator**

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.



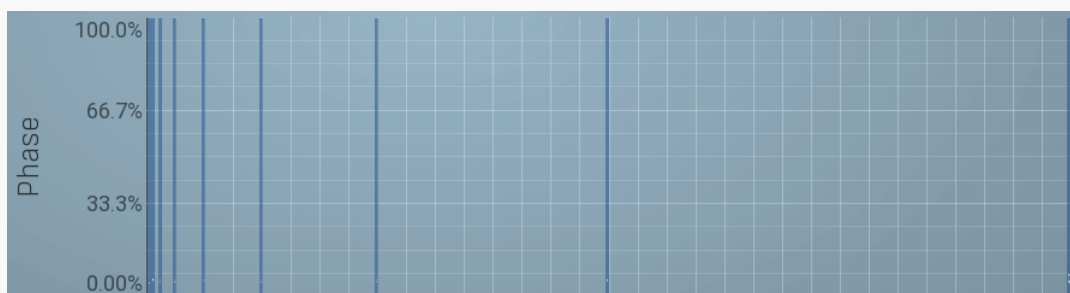
**Randomize**

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.

## Rate panel

RATE

Sync ?

Frequency

1.000 Hz

Sync group

Disabled

1

2

3

4

☐ Reset on ARP

Length

1 / 4

◀ ▶

Type

Straight

◀ ▶

Phase

90° [25.0%]

Count

1

Set frequency

Rate panel contains parameters controlling the speed of the LFO, whether the modulator is set to **Normal** mode or any other mode while the **LFO modulation** is used.

### Sync

Sync switch turns the modulator into synced mode, where its speed is not defined by frequency, but it uses musical units instead.

Frequency

1.000 Hz

### Frequency

Frequency defines the modulation speed.

Sync group

Disabled

1

2

3

4

### Sync group

Sync group lets you synchronize the modulators with each other and potentially with other parts of the plugin. It can be controlled only when **to-host synchronization** is disabled, otherwise it is overridden by synchronization from the host. By using the same synchronization group for all modulators you ensure they will always be in-sync even though no other synchronization is used. This can be useful, for example, when you want to modulate different parameters with different shapes or when using some more advanced method, such as using a follower. When the synchronization is enabled, it works on the 'first is the leader' basis, hence the first modulator controls the rest of the modulators in the same group.

☐ Reset on ARP

### Reset on ARP

Reset on ARP makes the LFO phase reset every time the ARP position jumps to the beginning. This is useful if you want to make the ARP and LFO in sync in a way.

## Synchronization panel

Length

1 / 4

◀ ▶

Type

Straight

◀ ▶

Phase

90° [25.0%]

Count

1

Set frequency

Synchronization panel contains parameters for the to-host synchronization.

Length 1 / 4 Length

Length defines the note length to be used.

Type Straight 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 90° (25.0%) Phase

Phase defines the phase offset of the to-host synchronization.

Range: 0° (0%) to 360° (100.0%), default 90° (25.0%)

Count 1 Count

Count defines the number of the units, hence multiplies of the sync length.

Range: 1 to 64, default 1

Set frequency Set frequency

Set frequency button sets the **Frequency** parameter available for the frequency mode so that it matches the current synchronization. That way you can set the modulator's frequency to the current synchronization and then change it a little for example.

## MIDI reset panel

### MIDI RESET - (RE)TRIGGER

Enable ?

Note-on Note-off Single shot  
Note-on only first Note-off only last Single shot reset  
Min velocity 0.00% Max velocity 100.0%  
Min note 0.000 Max note 127 (G9)  
Phase 0° (0%) Channel All

MIDI reset panel configures the MIDI reset feature, which will reset the oscillator when a MIDI note is received or its **MIDI reset parameter** is a target of another modulator or multiparameter. This way you can make the oscillator perform "in-sync" with your playing. Please note that once you enable it, the oscillator will not be in phase-sync with the host.

Enable Enable

Enable button enables or disables the feature.

Note-on Note-on

Note-on controls if the MIDI reset should occur when a note is pressed.

Note-off Note-off

Note-off controls if the MIDI reset should occur when a note is released.

Single shot Single shot

Single shot button activates the single shot mode in which the oscillator doesn't cycle around but instead only goes once from left to right, then stops until the MIDI reset occurs.

Note-on only first Note-on only first

Note-on only first controls if the MIDI reset should occur when a note is pressed only if it is the first note (thus no other note is being held).

Note-off only last Note-off only last

Note-off only last controls if the MIDI reset should occur when a note is released only if it is the last note (that is, no other note is being held afterwards).

Single shot reset Single shot reset

Single shot reset button defines if the phase should reset to 0 after a single shot period ends. For most waves such as sine it doesn't really matter since the value at 0 (the start of the cycle) is the same as value at 1 (the end of the cycle). But it might matter for saw wave for example.

Min velocity Min velocity

Min velocity defines the minimum velocity that will reset the oscillator.

Max velocity 100.0% **Max velocity**

Max velocity defines the maximum velocity that will reset the oscillator.

Min note 0.0 Hz **Min note**

Min note defines the minimum note that will reset the oscillator.

Max note 127.000 **Max note**

Max note defines the maximum note that will reset the oscillator.

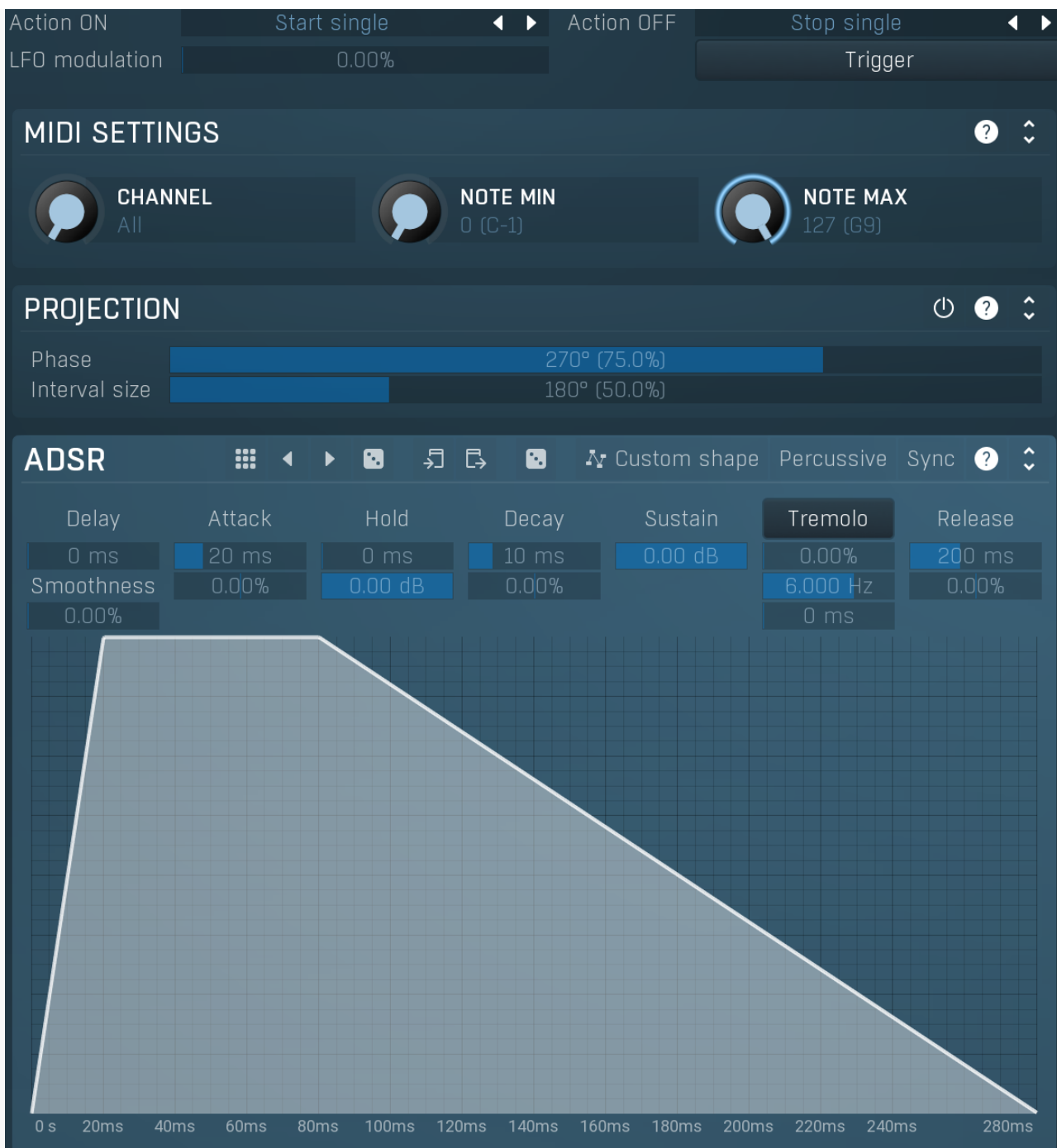
Phase 0° (0%) **Phase**

Phase defines the initial oscillator phase after a reset.

Channel All ◀ ▶ **Channel**

Channel defines note MIDI channel to reset the oscillator.

## Envelope mode



Envelope mode makes the modulator generate arbitrary envelopes from input MIDI or by analyzing the audio input level. When using MIDI the modulator responds to input note-on and note-off messages. When using an audio input (if available), the modulator detects the input level and when it exceeds the **Threshold On**, it behaves like a note-on MIDI event. Afterwards when the level drops down below **Threshold Off**, it behaves like a note-off MIDI event. Each event can result in just about any action. By default note-on starts the envelope and note-off initiates the release stage, but a different behaviour is also possible depending on **Action ON** and **Action OFF** parameters.

#### Action ON Start single Action ON

Action ON controls what happens when a note-on event occurs (either via audio or MIDI).

**Start single** action, which is the default, means that the envelope will start on the note-on event, but only once, it won't start again until you release all of the keys that are relevant for MIDI triggering only. With audio triggering it works the same as **Start** mode.

**Start** makes the envelope start every time you press a key, whether another key is already pressed or not. The envelope will seamlessly jump to the attack stage avoiding any abrupt changes.

**Start forced** is similar, but lets the envelope start from the very beginning every time. So for example if the envelope is currently in a long release stage and the new modulator value is 0.5, then the Start action jumps to a location in the attack stage where there is a value of 0.5 as well, hence avoiding abrupt changes. Start forced action on the other hand starts the whole envelope over from the beginning of the attack stage, where the value is most likely 0.

**Start legato** is in a way an opposite of the **Start single** in that it starts only if there is at least one other note already playing. As such it can only be used with MIDI triggering. With audio triggering it works the same as **Start** mode.

**Ignore** action simply ignores this event.

The remaining actions are rather creative and let you do the opposite - initiate release stage and stop the envelope when you press a key.

#### Action OFF Stop single Action OFF



Action OFF controls what happens when a note-off event occurs (either via audio or MIDI).

**Stop single** action, which is default, means that the envelope will enter the release stage on the note-off event, but only once, at the moment you release the last key (if you were holding more than one) it is relevant for MIDI triggering only.

**Stop** makes the envelope enter the release stage every time you release a key, whether another key is already pressed or not.

**Ignore** action simply ignores this event.

The remaining actions are rather creative and let you do the opposite - start the envelope when you release a key.

LFO modulation 0.00% **LFO modulation**

LFO modulation defines the amount of LFO modulation applied in addition to the envelope. With 0% the modulator uses only the envelope; with 100% the modulator does the same job as if the modulator were in **Normal** mode. To set the LFO parameters switch to normal mode temporarily.

Range: 0.00% to 100.0%, default 0.00%

Trigger

**Trigger**

Trigger button servers for manual triggering. It can be associated to other modulators as well for example, so it enables you to trigger the envelope pretty much any way.

## MIDI panel

### MIDI SETTINGS



CHANNEL

All



NOTE MIN

0 (C-1)



NOTE MAX

127 (G9)

MIDI panel contains parameters of the MIDI event detector.

## Projection panel

### PROJECTION

Phase

270° (75.0%)

Interval size

180° (50.0%)

Projection panel contains parameters of projection onto the LFO oscillator shape, which takes the value generated by the modulator and puts it onto the LFO oscillator shape. This features is useful for several creative effects.



**Enable**

Enable button enables or disables the projection onto the LFO oscillator shape.

Phase

270° (75.0%)

**Phase**

Phase defines the offset from zero of the signal curve. By default it is 75%, because when you look at common oscillator shapes, such as a sine or triangle, at position 75% its value is minimal. Then when you look at the right side, the value is growing up to the 25%, where it becomes the maximum.

Range: 0° (0%) to 360° (100.0%), default 270° (75.0%)

Interval size

180° (50.0%)

**Interval**

Interval defines the size of the interval from the oscillator shape in addition to **Phase**. As a result, phase defines where you start on the shape and interval specifies size of the window on the shape. Default value is 50% as for example sine grows from minimum to maximum in 50% of the period.

Range: 0° (0%) to 720° (200.0%), default 180° (50.0%)



**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.

200 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**

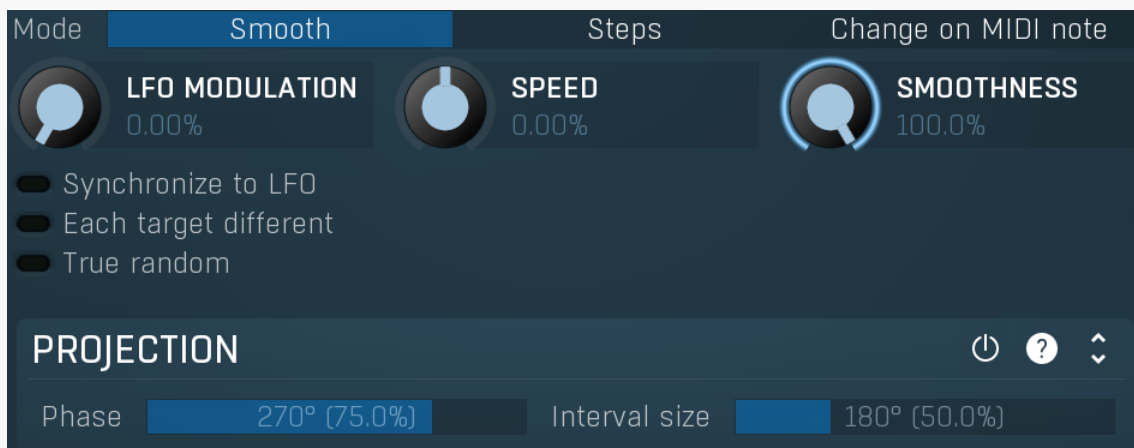
Smoothing lets you smoothen the entire envelope avoiding abrupt jumps. Note that in some cases involving short jumps the results may be a bit obscure.

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.

## Random mode



Random mode makes the modulator generate a pseudorandom sequence. Please note that despite its name, it is created so that it generates the same sequence every time. However the generator is linked to the **Speed** parameter, so if you change it, the whole sequence changes.

Mode

Smooth

Steps

Change on MIDI note

Mode

Mode defines the behaviour of the randomizer.

**Smooth** produces a continuous random modulation. **Smoothness** then controls how smooth it will be, where 0% means it will connect distinct values by straight lines, 100% means the modulation will be a completely smooth curve walking through these random points.

**Steps** produces a step change every particular time interval. It can also granularize it to a specified number of possible values according to **Smoothness** value. 100% disables the granularization. Otherwise the number of steps is the number of percentage values, so 3% means there will be 3 possible values, equally distributed over the range, let's call them 0%, 50% and 100%. Since it doesn't make sense to have 0 or 1 steps, the minimum is always 2. 2 steps essentially means the modulator is randomly switching between the minimum and maximum values for all associated parameters.

**Change on MIDI note** generates a random value every time a MIDI note is received by the plugin.



LFO MODULATION

0.00%

**LFO modulation**

LFO modulation defines the amount of LFO modulation applied in addition to the random generator. With 0% the modulator uses only the randomizer; with 100% the modulator does the same job as if the modulator were in **Normal** mode. To set the LFO parameters switch to normal mode temporarily.

Range: 0.00% to 100.0%, default 0.00%



SPEED

0.00%

### Speed

Speed defines the speed of the random changes proportional to the current tempo. 0% means that the speed is the same as your song's tempo.

Range: -1000.0% to 1000.0%, default 0.00%



SMOOTHNESS

100.0%

### Smoothness

Smoothness defines the amount of smoothing of the randomizer curve in order to minimize abrupt edges.

Range: 0.00% to 100.0%, default 100.0%

☒ Synchronize to LFO

### Synchronize to

#### LFO

Synchronize to LFO lets you synchronize the speed of the random sequence to LFO (Normal mode), hence also to your host. **Speed** is still applicable and, for example, +100% means 2x speed, +200% means 4x the speed etc.

☒ Each target different

### Each target

#### different

Each target different transforms value for each target, so that you can easily produce lots of different random values.

☒ True random

### True random

True random makes the modulator produce a true pseudo-random sequence independent of the current position within the project. By default this is disabled, so that every time you play your project, it sounds the same. But you might want to enable this option, for live performances for example.

## Projection panel

### PROJECTION



Phase

270° (75.0%)

Interval size

180° (50.0%)



### Enable

Enable button enables or disables the projection onto the LFO oscillator shape.

Phase

270° (75.0%)

### Phase

Phase defines the offset from zero of the signal curve. By default it is 75%, because when you look at common oscillator shapes, such as a sine or triangle, at position 75% its value is minimal. Then when you look at the right side, the value is growing up to the 25%, where it becomes the maximum.

Range: 0° (0%) to 360° (100.0%), default 270° (75.0%)

Interval size

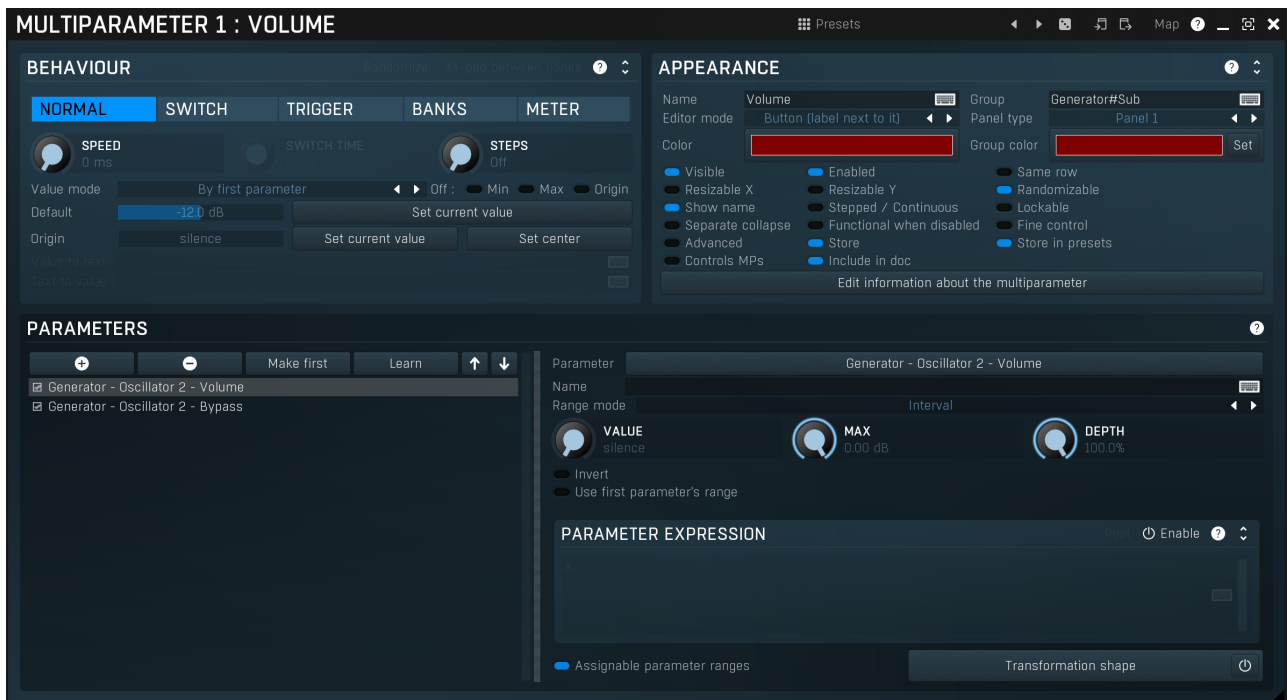
180° (50.0%)

### Interval

Interval defines the size of the interval from the oscillator shape in addition to **Phase**. As a result, phase defines where you start on the shape and interval specifies size of the window on the shape. Default value is 50% as for example sine grows from minimum to maximum in 50% of the period.

Range: 0° (0%) to 720° (200.0%), default 180° (50.0%)

# MultiParameter editor



MultiParameter is a powerful structure, which can speed up your workflow significantly and even perform automatic tasks, often useful when performing in real-time for example. Essentially a multiparameter is a controller which controls other parameters, in fact, an unlimited number of them. Each parameter has limits and potentially a transformation curve for more advanced processing. By manually moving the multiparameter (or automating/modulating it) you can control all of the associated parameters at once.

*This is just the beginning, but it is worth demonstrating how it could be used. We will show it on a vibrato effect. MVibratoMB (and partly MVibrato) is very good at simulating rotary speakers. A rotary speaker traditionally contains a speed switch, or in our case we will think of it as a speed knob - a control that alters the spin speed of the rotary. This would normally be the **Rate** parameter of the vibrato. However, when the rate is increased, the vibrato starts changing the pitch too much, sounding a little too "honky-tonk". We can compensate for this by lowering the **Depth** parameter. As it is not very convenient to control 2 parameters at once, we use a multiparameter to control both parameters with appropriate ranges (ascending for the **Rate** and descending for the **Depth**).*

Besides this basic usage, multiparameters can also work as triggers and switches. Set a multiparameter's mode to **Trigger** or **Switch** and it stops being a slider and becomes a button. When you click the button, the multiparameter starts moving on its own - over the dialled-in switch time it will increase its value (and also the values of any associated parameters) to a maximum and, in the case of trigger mode, then decrease it back to a minimum. In switch mode clicking the button again, the multiparameter decreases back to the minimum value. To make the multiparameter into a simple switch, we can set the switch time to minimum, but in this case we want to extend the functionality in our rotary example.

*As mentioned, rotary speakers often have a speed switch. Once switched on, the speed starts increasing until it reaches the "fast" setting, and when switched off, the speed starts decreasing to the original "slow" rate. All we need to do to replicate this functionality is to set the multiparameter's mode to 'switch'.*

*A real rotary actually has 2 speakers, one for low frequencies and the other for the higher ones. As you might expect, these do not have the same spin rate nor do they speed up or slow down equally either. Here is where we can start showing the true potential of multiparameters.*

*To simulate this, we have to use two bands of MVibratoMB, the first one will simulate the lower reproducer, and the second will be the higher. We use the first multiparameter to control the first band's rate in the same way as described in the example above. Similarly, we use the second multiparameter to control the second band's rate. Now we have 2 switches and can make each band speed-up or slow-down separately, but we want just one switch for both bands. To do this, we use a third multiparameter to control the first and second multiparameters, in switch mode again but with a 0ms switch time. Pressing the button of the 3rd multiparameter instantly activates the other 2 multiparameters, they both start speeding-up, over a different time period as we requested. Pressing the button again, releases it which also instantly releases the first 2 multiparameters and they start slowing down. Just like the real thing.*

Now that we have shown you what is possible with multiparameters, it is worth mentioning that they are used extensively for building devices on the easy screens of most Melda plugins. Every multiparameter given a name in the **Information** panel will be shown on the Easy screen (if the plugin has one). Check our online video tutorials to get more information about **multiparameters and building devices**.

It is also worth mentioning that you can access the multiparameter settings directly from easy screen by holding Ctrl+Alt and clicking on the target control. It may simplify building devices. Note that this may not work for some editor modes such as meters or bar graphs.

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.



**Map**

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

Behaviour

BEHAVIOUR

Randomizer - XY-pad between banks

NORMAL

SWITCH

TRIGGER

BANKS

METER

SPEED

0 ms

SWITCH TIME

0 ms

STEPS

Off

Value mode

By first parameter

◀ ▶

Off : 

Min

Max

Origin

Default

-12.0 dB

Set current value

Origin

silence

Set current value

Set center

Value to text

Text to value

**Randomizer**

Randomizer switch is available only for **Trigger** mode and it makes the multiparameter produce random values for each associated parameters. This is useful to implement some sort of randomization feature, which covers a set of parameters. You usually want to set the **Switch time** to 0, so that the randomization is instant, but longer values may be useful for some creative effects.

**XY-pad between banks**

XY-pad between banks switch is available only for Banks mode and it lets you create XY pads, that would interpolate between 3 or more banks that you specify. With 4 banks the engine creates a classic XY pad, where the 1st bank belongs to the left top corner, 2nd to the right top, 3rd to left bottom and 4th to the right bottom. With more banks the engine creates a circular pad with vertices associated to individual banks.  
Please note that in order for this to work, the multiparameter actual needs 2 multiparameters (X and Y values), hence must NOT be the last one and it occupies the next multiparameter as well. It is recommended to name the next multiparameter and associate it to some parameters, ideally the same ones, just to make sure the engine won't remove it. But in fact only the first multiparameter will actually be working.

NORMAL

SWITCH

TRIGGER

BANKS

METER

Mode

Mode controls the behaviour of the multiparameter.

**Normal** mode makes the multiparameter work like any other control.

**Switch** mode hides the slider and shows a button instead. The button has 2 states. By pushing the button, the multiparameter value starts rising from 0% to 100% over a specified time interval. By pushing it again the value starts falling back to 0%. You could do the same thing having the multiparameter in normal mode and moving the slider from left to right and then back, but mode this performs that automatically and maintains a constant time period.

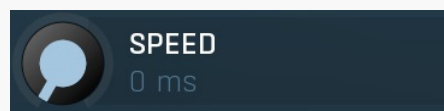
**Trigger** mode is similar to switch mode, but the button has only a single state and when you push it, the value automatically goes from 0% to 100% and then back without any need to push the button again.

**Banks** mode is very different. A multiparameter in banks mode keeps several states (called banks) for all of the parameters, much like A-H presets, but only with a limited set of parameters. The multiparameter then morphs between the banks or can be set to switch directly between them (no interpolated values). This is a marvellous way to control many parameters with complex settings by using a single multiparameter.

Let's explain the banks mode in more detail. Say you switch a multiparameter to banks mode, learn a few parameters and set the number of banks to 4. Then bank 1 contains a value for all of the parameters. Similarly bank 2 contains a different value for each of them. And so on. If you set the multiparameter slider to 0%, the associated parameters will be set to values in bank 1. If you set the slider to 100%, bank 4 will be used. If you set the slider to 33.3%, bank 2 will be used. And what if you select 50%? Then it will be halfway between bank 2 and bank 3.

You can have many banks, you can edit each of them, generate random settings etc. So let's say you want to create some complex movement. You use a multiparameter in banks mode, select a reasonable number of banks. You can edit each of them, but it is easier to use the randomization button to generate random settings for each of them. Then every time you move the multiparameter, all of the associated parameters will move, somewhere between the banks. You can then use a modulator or automation to slowly adjust the multiparameter.

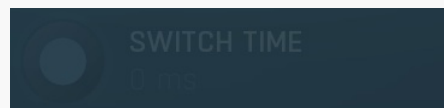
**Meter** mode makes the multiparameter work as a meter. Instead of controlling other parameters it starts following the value of them. You can then use that to implement a simple meter on the easy screen (if the plugin has one).



### Speed

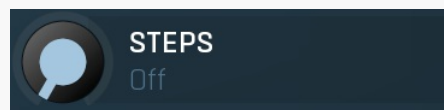
Speed controls the interpolation time. When it is zero and you change the multiparameter value, the associated parameters are adjusted immediately. If this is non-zero however, the actual parameters won't change immediately but will interpolate over time. The speed value is actually the time needed to go from minimum to maximum or vice versa. So if this is 1 second and the current value is say 0% and you click 100%, it will take 1 second for the multiparameter to get there.

This feature is provided mainly because changing some parameter via MIDI or mouse may cause unnecessary zipper noise or inaccuracies due to low MIDI precision. Using the interpolation you can somewhat slow everything down, so that the artifacts become negligible. It can also be used creatively. The default value has been experimentally tested to avoid all artifacts for most parameters.



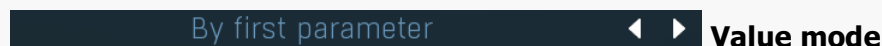
### Switch time

Switch time defines the time needed to switch from the minimum value to the maximum one, or conversely. It is used only in **switch** and **trigger** modes.



### Steps

Steps lets you create an arbitrary number of equi-distant steps for the multiparameter values. While this technically limits the possibilities of the multiparameter by limiting the number of accessible values, it is sometimes easier to choose from a predefined number of options than from the full range. If you want to use different ranges between the steps, use the Banks mode with Interpolate values disabled.



Value mode defines the units displayed on the multiparameter.

**Percents** mode lets the multiparameter display percentages between 0% to 100%.

**Percents (-100% to 100%)** displays percentages between -100% to 100%.

**By first parameter** mode uses the current value of the first parameter that is controlled by the multiparameter. For example, if you want to control a plugin gain, but also in addition to the changed gain control other parameters, you may still want to call the multiparameter "gain" and the units should be decibels as usual, not percentages which do not make much sense for such a multiparameter.

**By bank name** displays the name of the nearest bank. In some controls, such as switchers, it is possible to display the set of the values as a menu. The menu is created automatically and it even creates groups for better clarity, based on the prefix of the bank names. You can use ' # ' (hashtag surrounded by spaces) to define the groups manually like this: "group # name". For example, you can name one bank "Main group # First bank" and another "Main group # Second bank", and these will be displayed in a single group in the menu. If you are going to use this method, make sure the ' # ' sequence is present in each bank's name.

**By bank name interpolated** considers name of all banks numbers. It then interpolates between them and displays the result as a number.

**By bank name interpolated log** is similar, but interpolates the values in logarithmic domain. considers name of all banks numbers. It's



useful for units, which are naturally logarithmic, such as frequency.

**By bank number** shows the index of the nearest bank.

**Expression** lets you formulate the value -> text and back using mathematical expressions and can be used for some more complex MPs or if you need some custom units.

Default -12.0 dB **Default**

Default controls the default value of the multiparameter. You can edit it directly or just set the MP into its reasonable default and click the **Set current value**. Most GUI components created for the multiparameter respond to right-click by setting the default value in the same way that other parameters do. It is essential for user experience when building your own devices.

Set current value **Set current value**

Set current value stores the current value as the default one for the multiparameter.

Origin silence **Origin**

Origin informs the GUI engine of the origin of the value. For instance, a default value for panorama is in the center and it is logical that visual elements controlling panorama should somehow highlight the center position. If, for example, you are using a value button to edit the panorama, by default it displays the current value using a bar starting from the left side (being the origin defined as minimum) towards the actual value, but here it is better to display the bar from the center towards the current value, whether it is on the left or right of the center. Therefore the center should be the origin.

Set current value **Set current value**

Set current value stores the current value as the origin for the multiparameter.

Set center **Set center**

Set center sets the center (50%) as the origin for the multiparameter. This is often the case for parameters such as gain and panorama, so it deserves a dedicated button. It is supported only for knobs.

Value to text

## Value to text

Value to text is available only with Expression **Value mode** and lets you enter a text featuring mathematical expressions to produce the units. This can be used for more complex multiparameters. The text itself is just another text, but it can contain substrings of this structure:

**{expression}**

or

**{expression;decimals}**

where expression is the actual mathematical expression and decimals is the number of decimal places in the resulting value. Here is the list of variables available for each expression:

**x** = the multiparameter value in 0..1

**sr** = current sampling rate

Functions and features of the expressions are available below. Now let's see a few examples:

*Test: {x}* - produces "Test: 0.12" (with MP value 0.1234)

*{x} ({x\*100;0}%)* - produces "0.12 (12%)" (with MP value 0.1234)

*{ffrom01(x);1} Hz* - produces say "215.56 Hz" ("20.0 Hz" for MP value 0 and "20000.0 Hz" for MP value 1)

*{todB(sqr(x));4} dB* - produces "-12.0412 dB" for 0.5. "sqr" is used to mimic the transformation used in pretty much every Volume parameter in MeldaProduction plugins.

Expression evaluator uses traditional C/C++ style formatting, which is natural for most people. It provides arithmetics, logical and conditional operators. Following terms are supported:

Constants: **pi**, **e**, **sqr2**, **ln2**

Arithmetic operators:

**-a** inverts the sign, e.g. "-x" produces +2 for x=-2

**a+b** = addition

**a-b** = subtraction

**a\*b** = multiplication

**a/b** = division

**a%b** = modulo, remainder after division

**a^b** = power, e.g. "2^3" produces 2\*2\*2 = 8

Arithmetic functions:

**min(a,b)** = minimum of both values

**max(a,b)** = maximum of both values

**limit(a,min,max)** = a limited into the interval min..max

**to01(a,min,max)** = converts "a" as min..max to 0..1

**from01(a,min,max)** = converts "a" as 0..1 to min..max

**tom11(a,min,max)** = converts "a" as min..max to -1..1

**fromm11(a,min,max)** = converts "a" as -1..1 to min..max



Basic mathematic functions: **abs(x)** = absolute value, e.g.  $\text{abs}(-3) = 3$  **sqr(x)** =  $x \times x$  **sqrt(x)** = square root **exp(x)** = natural exponential  $e^x$  **ln(x)** = natural logarithm **log10(x)** = logarithm with base 10 **log(x, base)** = logarithm with specified base **inv(x)** =  $1/x$  **sgn(x)** = sign of x, -1 or 0 or +1 depending on  $x \times x$  **round(x)** = rounding to the nearest value **floor(x)** = rounding to the nearest lower value, e.g.  $\text{floor}(-2.3) = -3$  **ceil(x)** = rounding to the nearest higher value, e.g.  $\text{ceil}(-2.3) = -2$  **rand(x)** = random value from 0 to x

Functions for specific units:

**f01(a)** = converts "a" as frequency from 20...20000 into log scale 0..1

**ffrom01(a)** = converts "a" as 0..1 (log scale) to frequency from 20...20000

**todb(a)** = converts "a" as multiplier to dB value by calculating  $20 \times \log_{10}(a)$

**fromdb(a)** = converts "a" as dB value to multiplier by calculating  $10^{(a/20)}$

Trigonometric functions: **sin(x)**, **asin(x)**, **cos(x)**, **acos(x)**, **tan(x)**, **atan(x)**, **sinh(x)**, **cosh(x)**, **tanh(x)**

Logical operators:

**a==b** = comparison producing 1 if "a" and "b" are equal, 0 otherwise

**a!=b** = comparison producing 1 if "a" and "b" are NOT equal, 0 otherwise

**a<b** = comparison producing 1 if "a" is lower than "b", 0 otherwise

**a<=b** = comparison producing 1 if "a" is lower or equal to "b", 0 otherwise

**a>b** = comparison producing 1 if "a" is greater than "b", 0 otherwise

**a>=b** = comparison producing 1 if "a" is greater or equal to "b", 0 otherwise

**!a** = logical negation, 0 produces 1, 0 otherwise

**a&& b** = logical AND, produces 1 if both "a" and "b" are nonzero

**a | b** = logical OR, produces 1 if any of "a" and "b" are nonzero

**a^b** = logical XOR, produces 1 if "a" and "b" are logically different

**a ? b : c** = if a is nonzero, then the result is b, otherwise it is c}}

## to value

Text to value is available only with Expression **Value mode** and lets you convert a number user enters as a text input into the multiparameter value in 0..1. This can be used for more complex multiparameters. Unlike **Value to text** here you need to enter a single expression, no need for any other text. If the resulting value exceeds the 0..1 interval, it is automatically limited. Here is the list of variables available for the expression:

**x** = the value the user entered

**sr** = current sampling rate

Functions and features of the expressions are available below. Now let's see a few examples:

**x** - sets MP to 0.1234 if the user entered "0.1234"

**x/100** - sets MP to 0.1234 if the user entered "12.34" (or "12.34%" for example)

**f01(x)** - sets MP to the proper frequency in log scale, e.g. 20 is translated to 0, 20000 to 1

**fromdb(sqrt(x))** - sets MP to 0.5 if the user entered "-12.0412" (or "-12.0412 dB" for example). "sqrt" is used to mimic the transformation used in pretty much every Volume parameter in MeldaProduction plugins.

Expression evaluator uses traditional C/C++ style formatting, which is natural for most people. It provides arithmetics, logical and conditional operators. Following terms are supported:

Constants: **pi**, **e**, **sqrt2**, **ln2**

Arithmetic operators:

**-a** inverts the sign, e.g. "-x" produces +2 for  $x=-2$

**a+b** = addition

**a-b** = subtraction

**a\*b** = multiplication

**a/b** = division

**a%b** = modulo, remainder after division

**a^b** = power, e.g. "2^3" produces  $2 \times 2 \times 2 = 8$

Arithmetic functions:

**min(a,b)** = minimum of both values

**max(a,b)** = maximum of both values

**limit(a,min,max)** = a limited into the interval min..max

**to01(a,min,max)** = converts "a" as min..max to 0..1

**from01(a,min,max)** = converts "a" as 0..1 to min..max

**tom11(a,min,max)** = converts "a" as min..max to -1..1

**fromm11(a,min,max)** = converts "a" as -1..1 to min..max

Basic mathematic functions: **abs(x)** = absolute value, e.g.  $\text{abs}(-3) = 3$  **sqr(x)** =  $x \times x$  **sqrt(x)** = square root **exp(x)** = natural exponential  $e^x$  **ln(x)** = natural logarithm **log10(x)** = logarithm with base 10 **log(x, base)** = logarithm with specified base **inv(x)** =  $1/x$  **sgn(x)** = sign of x, -1 or 0 or +1 depending on  $x \times x$  **round(x)** = rounding to the nearest value **floor(x)** = rounding to the nearest lower value, e.g.  $\text{floor}(-2.3) = -3$  **ceil(x)** = rounding to the nearest higher value, e.g.  $\text{ceil}(-2.3) = -2$  **rand(x)** = random value from 0 to x

Functions for specific units:

**f01(a)** = converts "a" as frequency from 20...20000 into log scale 0..1

**ffrom01(a)** = converts "a" as 0..1 (log scale) to frequency from 20...20000

**todb(a)** = converts "a" as multiplier to dB value by calculating  $20 \times \log_{10}(a)$



Text

**fromdb(a)** = converts "a" as dB value to multiplier by calculating "10<sup>^(a/20)</sup>"

Trigonometric functions: **sin(x)**, **asin(x)**, **cos(x)**, **acos(x)**, **tan(x)**, **atan(x)**, **sinh(x)**, **cosh(x)**, **tanh(x)**

Logical operators:

**a==b** = comparison producing 1 if "a" and "b" are equal, 0 otherwise

**a!=b** = comparison producing 1 if "a" and "b" are NOT equal, 0 otherwise

**a<b** = comparison producing 1 if "a" is lower than "b", 0 otherwise

**a<=b** = comparison producing 1 if "a" is lower or equal to "b", 0 otherwise

**a>b** = comparison producing 1 if "a" is greater than "b", 0 otherwise

**a>=b** = comparison producing 1 if "a" is greater or equal to "b", 0 otherwise

**!a** = logical negation, 0 produces 1, 0 otherwise

**a&&b** = logical AND, produces 1 if both "a" and "b" are nonzero

**a|b** = logical OR, produces 1 if any of "a" and "b" are nonzero

**a^^b** = logical XOR, produces 1 if "a" and "b" are logically different

**a ? b : c** = if a is nonzero, then the result is b, otherwise it is c}}

## Appearance

**APPEARANCE**

Name: Volume    Group: Generator#Sub

Editor mode: Button (label next to it)    Panel type: Panel 1

Color: [Red color picker]    Group color: [Red color picker] Set

☒ Visible    ☒ Enabled    ☐ Same row

☐ Resizable X    ☐ Resizable Y    ☒ Randomizable

☒ Show name    ☐ Stepped / Continuous    ☐ Lockable

☐ Separate collapse    ☐ Functional when disabled    ☐ Fine control


☐ Advanced    ☒ Store    ☒ Store in presets

☐ Controls MPs    ☒ Include in doc



Edit information about the multiparameter

Name Volume  **Name**

Name specifies the name of the multiparameter, which is shown on the multiparameter button. The name is also used for devices - the multiparameter serves as a parameter for the device (on the Easy screen). If no name is specified or if the first character is an \*, then the parameter is hidden. This is useful if you need some internal multiparameters which you don't want to show on the Easy screen for some reason.

Group Generator#Sub  **Group**

Group can be used to put some multiparameters into the same group, which results in them being placed in the same panel on the Easy screen (the device editor). Additionally you can actually place the groups into tabs by setting group to "tabname#groupname". The name of the tab needs to be there only for the first parameter of the new group. This makes it possible to build a complex devices with dozens of parameters.

Editor mode Button (label next to it)   **Editor mode**

Editor mode controls the way the multiparameter are to be displayed on the Easy screen.

**Normal** is the default mode and is represented by a small knob or button.

**Big** mode is similar, but uses a big knob or big button.

**Button** mode displays a value button, which is usually more compact than knobs.

**Check-boxes** makes the multiparameter displayed as a set of checkboxes (also called radio buttons). It is relevant only in **Banks** mode.

**Check-boxes horiz & below** is similar but displays the checkboxes in a single row, hence horizontally. Below mark makes the label underneath the actual checkbox.

**Switcher and Selectors** are useful for selecting a number of discrete values and similarly to check-boxes these are working only in **Banks** mode.

**Title button** places the control into the title bar of the panel to which it belongs.

**Title enable button** places the control into the title bar of the panel to which it belongs and makes it a standard enable button (which also makes all controls within the panel unavailable if it is itself disabled).

**XY pad** creates a 2 dimensional XY pad control, that edits this multiparameter in the X axis and the next multiparameter in the Y axis. There are multiple versions of this control, all of them differ only by size.

**Spacer** is a helper mode for device design, which doesn't display anything and only keeps empty space.

**Meter** creates a simple meter instead. You will probably want to set the multiparameter to Meter mode as well or attach it to a

modulator. Meters don't really control anything and their purpose is purely to get a visual feedback. The meters can be horizontal or vertical and they can be up or down. Up is the usual choice useful for peak meters for example. Down is useful for gain reduction meters.

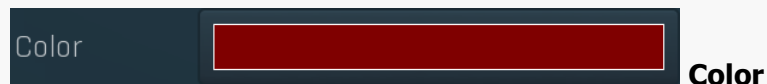
**Bars start/end** mode creates an editor, similar to step sequencer editor, where each parameter has its own bar. The **Bars start** starts the editor and all multiparameters are then added to it until a multiparameter with **Bars end** mode is found or until there are no remaining multiparameters. Note that this kind of editor doesn't show units and may have several other limitations.

**Order** is a very specific editor for Order modules available in modular systems such as MXXX. It lets you provide an processing order editor on the easy screen. To use it, attach the MP to Order parameter of the Order module and edit the MP information field, so that it contains all the items to be ordered, separated by ';'. The number of items must match the number of items in the Order module, otherwise the order won't work properly. You can also include colors for each item separated by # (hexa or one of the predefined set: Dynamics, Distortion, Modulation, Stereo, Spectral, Synthesis, Instrument, MDrummer, Reverb, Delay, EQ, Filter, Saturation, Limit, Time, Pitch, FX) and enable MP indices.

Example of the info: Compressor#Dynamics;EQ#EQ;Limiter#007F7F;Something

## Panel type Panel 1 ◀ ▶ Panel type

Panel type defines the type of panel in which multiple controls of the same group are placed. These differ only in their graphics display.



Color defines colorization for the element on the Easy screen (if the plugin has one). The feature is disabled if the Alpha value of the color is 0. Using this feature often increases memory consumption of the plugin, so make sure you use it only if necessary and try to use as low a number of different colors as possible. It is recommended to use only the snapshot colors to make sure the same colors are used in most cases, reducing the memory consumption. It is also highly recommended to use colors with a value (lightness) of 128 (the middle value), which makes sure that the lightness of the elements won't be changed. This works best for most styles. Please note that the style may be configured to simply ignore this color, so there may be no change at all. If you use this feature, make sure that you test it with all styles.

For the sake of workflow the colors have predefined meanings. It's highly recommended to follow this standard:

- Orange - dynamics
- Green - equalization, filtering
- Brown/yellow - reverb, delay
- Blue - modulation
- Red - limiting, saturation, distortion
- Cyan/yellow - stereo
- Purple/pink - time, pitch, unison...
- Grey - utilities, tools

## Group color Set Group color

Group color defines colorization for the group panel on the Easy screen (if the plugin has one) and is ignored for all multiparameters except for the first one in a group. The feature is disabled if the Alpha value of the color is 0. Using this feature often increases memory consumption of the plugin, so make sure you use it only if necessary and try to use as low number of different colors as possible. It is recommended to use only the snapshot colors to make sure the same colors are used in most cases, reducing the memory consumption. It is also highly recommended to use colors with a value (lightness) of 128 (the middle value), which makes sure that the lightness of the elements won't be changed. This works best for most styles. Please note that the style may be configured to simply ignore this color, so there may be no change at all. If you use this feature, make sure you test it with all styles.

For the sake of workflow the colors have predefined meanings. It's highly recommended to follow this standard:

- Orange - dynamics
- Green - equalization, filtering
- Brown/yellow - reverb, delay
- Blue - modulation
- Red - limiting, saturation, distortion
- Cyan/yellow - stereo
- Purple/pink - time, pitch, unison...
- Grey - utilities, tools

## Set Set

Set button sets the color and group color for all multiparameters in the same group. It is pretty sensible to do that as all controls should look similar within each group. This can also be done by editing each parameter, but this way is easier.

## ☒ Visible Visible

Visible checkbox controls if the parameter is visible on the Easy screen (if the plugin has one). Its effect is similar to the '\*' prefix in the parameter name, but the multiparameter's name is also available to the plug-in host. This is useful when you wish to automate that multiparameter from the host but not show it on the Easy screen. This parameter can also be attached to another multiparameter for example in order to change the GUI somehow.

## ☒ Enabled Enabled

Enabled switch enables/disables the multiparameter. If disabled, it is grayed on the easy screen.

## ☒ Same row Same row

Same row checkbox defines if the parameter should be displayed next to the previous one on the Easy screen. Otherwise it will be

placed on the next row. This setting serves as a hint and the plugin may ignore it, if it is impossible to do.

### ☐ Resizable X **Resizable X**

Resizable X switch lets you specify if the panel could be resized. It is on by default to make sure everything gets resized, however when using multiple panels next to each other, it may be advantageous to disable resizing of some of them to save space. Otherwise each panel's size is proportional to number of controls it contains, which could make some of the panels larger than actually necessary.

### ☐ Resizable Y **Resizable Y**

Resizable Y switch lets you specify if the panel could be resized vertically. It is off by default to make sure everything has the minimum size it requires, but for aesthetic reasons you may want to make all groups on the same row the same size even if the controls inside them are not.

### ☒ Randomizable **Randomizable**

Randomizable option defines if the multiparameter can be randomized on the easy screen. You may want to disable this for input/output gain for example.

### ☒ Show name **Show name**

Show name option lets you show or hide the name of the multiparameter for some editor modes. The option has no effect for several editor modes.

### ☐ Stepped / Continuous **Stepped / Continuous**

Stepped / Continuous option tells the engine that the multiparameter can be in 2 modes, stepped or continuous. If so, it is assumed that you either used **Banks mode** or **Steps** to produce some sort of predefined set of values for the stepped mode. By enabling this option you allow the engine to convert the multiparameter to continuous mode by either ignoring the steps or interpolating the bank values. It can be used when designing devices.

### ☐ Lockable **Lockable**

Lockable option creates a lock button next to the parameter on the Easy screen, allowing the user to browse through presets without this parameter changing. Please note that this feature is available only for some editor modes.

When the parameter is first locked on the Easy screen it is added to the set of lockable parameters (which are listed in the Global Lock window).

### ☐ Separate collapse **Separate collapse**

Separate collapse checkbox makes the panel collapsable separately on the Easy screen. By default it is disabled and that makes the engine find all panels on the same row and collapse all of them or none of them.

### ☐ Functional when disabled **Functional when disabled**

Functional when disabled switch makes the multiparameter work even when disabled. This may be useful in some complex scenarios, where you need to make the MP control the target parameters and only use the **Enabled** flag to grey out the controls on the easy screen.

### ☐ Fine control **Fine control**

Fine control switch makes the multiparameter editor steps extra small, which is useful, when you need very high precision. This is often handy when using banks mode with many banks interpolating.

### ☐ Advanced **Advanced**

Advanced switch makes the multiparameter 'advanced', disabled by default. If such an multiparameter exists, the engine also creates a button to show/hide all advanced multiparameters. In effect this lets you create a simple GUI with optional advanced controls.

### ☒ Store **Store**

Store makes the multiparameter stored. Relevant when creating devices. In some rare cases you may want to disable this, e.g. if the value is a meter or somehow it just doesn't need to be save at all.

### ☒ Store in presets **Store in presets**

Store in presets makes the multiparameter stored in presets. Relevant when creating devices. In some cases you may want to store the value, but ignore it when browsing presets. For instance, input gain may be ignored if the user is supposed to set the gain to some predefined value and then browse the presets without changing that gain.

### ☐ Controls MPs **Controls MPs**

Controls MPs informs the engine that this MP is controlling other multiparameters. As such it shouldn't be 'invoked' unless it is explicitly touched by the user. This is relevant for example when it is used in a device with a device presets - loading a device preset invokes all relevant MPs, so that the state of the device is remembered properly, but invoking this one would actually make things incorrect.

### ☒ Include in doc **Include in doc**

Include in doc makes the multiparameter's help included in the documentation / panel help. You may want to disable this if there are multiple 'identical' controls with the exact same help info, for all of them except for one.

Edit information about the multiparameter

Edit

## information about the multiparameter

Edit information about the multiparameter lets you edit the information displayed as help for the multiparameter control on the easy

screen. It can use a simplified HTML format, but in practice only following tags are likely to be useful:

**<name>** = name of the control, the info should begin with that.

**<reference>** = reference to another control.

**<value>** = specific value the control can have.

**<b>** = bold text

**<i>** = italic text

**<cred>** = red

**<cgreen>** = green

**<cblue>** = blue

For example: `<name>Gain</name> controls the output level. It can be used in conjunction with <reference>Volume</reference>. When set to <value>silence</value>, it won't do anything and consume no CPU power.`

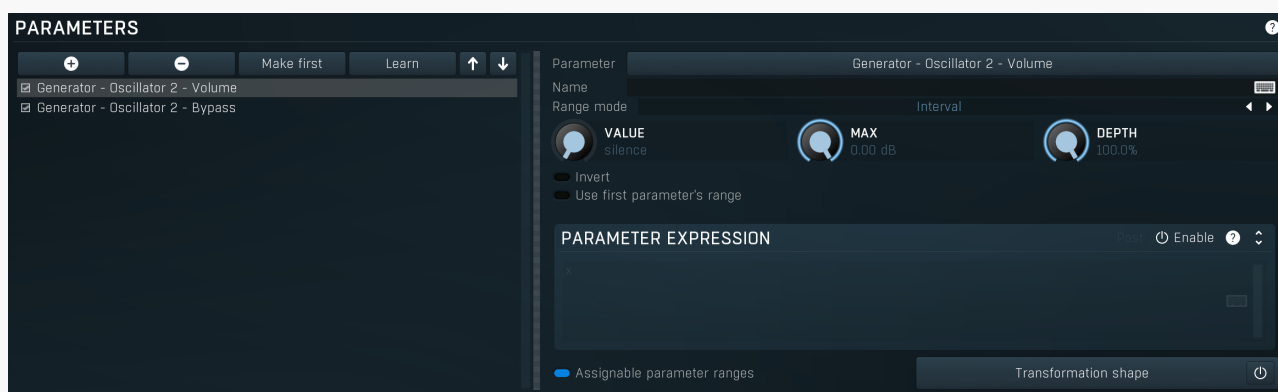
When **Order** control is used, the first line of the information can contain additional info of ";" separated individual items in the following format:

`name{#hexcolor}{*index of the enable MP}` where {} denotes an optional item.

For example: `Reverb;EQ#7F0000;Flanger#007F00*3`

In this example, the order is probably used to control order of effects. When the order MP value is set to 0, Reverb is the first, with no specific color and no MP to enable/disable it. Flanger is next, the color is red, but no enable either. Finally Flanger is the third, it is green and MP3 is used to enable/disable it.

## Parameters panel



Parameters panel configures how the multiparameter assigns values to the target parameters.



### Add

Add button adds a parameter to the list of controlled parameters. Alternatively you can use the learn feature available by right-clicking the multiparameter button.



### Delete

Delete button deletes the selected parameter from the list of controlled parameters.

Make first

### Make first

Make first button moves the selected parameter to the first item in the list. This is useful for sake of the **By first parameter** value mode, which makes the multiparameter show the units of the first parameter in the list. Please note that if you have some other multiparameter, modulator or another subsystem access the ranges of individual parameters, this function will reorder them, so these connections will no longer be correct.

Learn

### Learn

Learn button starts or stops the learning. Click it, then move some parameters in the plugin, then click it again. Learning can also be accessed from the global multiparameter menu.



### Up

Up button moves the selected parameter up one item, if possible. This may be useful when keeping things organized, but please note that if you have some other multiparameter, modulator or another subsystem access the ranges of individual parameters, this function will reorder them, so these connections will no longer be correct.



## Down

Down button moves the selected parameter down one item, if possible. This may be useful when keeping things organized, but please note that if you have some other multiparameter, modulator or another subsystem access the ranges of individual parameters, this function will reorder them, so these connections will no longer be correct.

Parameter

Generator - Oscillator 2 - Volume

## Parameter

Parameter defines the target parameter which is being modulated. The set contains all automatable parameters.

Name



## Name

Name lets you name the parameter somehow and may be helpful in situations, where there are many parameters being edited without obvious meanings.

☒ Assignable parameter ranges

## Assignable parameter ranges

Assignable parameter ranges allows you to assign parameter ranges of several first parameters to other subsystems such as multiparameters or modulators. By default it is disabled, which removes all the relevant parameters to save valuable resources. This feature is available only if automation compatibility mode for V10 is disabled.

Transformation shape



## Transformation shape

Transformation shape button displays the graph editor, which lets you tweak the shape of the curve used to control the selected parameter. The X axis shows the original values, the Y axis defines the results. Please note that this takes some CPU, therefore you have to enable it using the enable button in the title bar.

Range mode

Interval



## Range mode

Range mode defines how the parameter range is selected. While sometimes it is better to specify minimum and maximum, other times it is better to use a nominal center and depth (% of full scale). This control allows you to define which one it will be.

**Up and down** mode makes the values go above and below the selected **Value**, which is considered the center. The interval is made smaller if necessary.

**Full range mode** is similar, except the range is symmetrically constrained, so the selected **Value** may not be the center anymore.

**Up/down only modes** goes from the selected value up/down only.

Let's compare these 4 modes. Taking a value of -12dB value, with a depth of 75% and a scale of +/- 24dB. The nominal range is therefore = +/-24 dB \* 75% = 36dB. With values of 0%, 50% and 100% the outputs are:

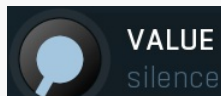
Up and down: -24, -12, 0 (range constrained to 12 dB either side)

Full range: -24, -6, 12 (range limited to minimum, but not constrained)

Up only: -12, 6, 24 (range not constrained = +/-24 dB \* 75% = 36dB)

Down only: -12, -18, -24 (range limited to minimum)

**Interval mode** is the most simple one and goes from **Value** to **Maximal value**.



## Value

Value defines the center of the target parameter's range or the minimum if the **Range mode** is set to **Interval**.



## Maximal value

Maximal value defines the upper limit of the target parameter's range. It is available only if the **Range mode** is set to **Interval**. This value can be lower than **Value**. 0% is always mapped to reference>Value and 100% to reference>Maximal value.



## Depth

Depth defines size of the target parameter's range. It is used only if the **Range mode** is not set to **Interval**.

☐ Invert

## Invert

Invert checkbox inverts the target parameter's range, so that minimum becomes maximum and vice versa.

☐ Use first parameter's range


Use

## first parameter's range

Use first parameter's range makes the parameter display use the same range as the first parameter in the list. This is often useful if want to control the range in some way and apply the range to multiple parameters.

# Parameter expression

## PARAMETER EXPRESSION

Post  Enable  

x




Parameter expression panel lets you provide an algebraic expression for computing the final parameter value. Here are several predefined variables you can use in your expressions (# denotes a positive integer number):

**x** contains the current multiparameter value (in **Meter mode** it contains the parameter value);

**p#** contains the current value of the parameter number # in the **parameter list** (e.g. "p2+p3" computes the sum of values of parameters 2 and 3 in the **parameter list**);

**mp#** contains the current value of the multiparameter number #.

For example, an expression "(x+p2+mp5)/3.0" calculates the average of the values of the current multiparameter, parameter number 2 in the **parameter list**, and the multiparameter with index 5.



### Post

Post button toggles whether the transform and min/max range are applied to the value x before evaluating the expression (Postprocessing mode, button on), or to the result of evaluating the expression (Preprocessing mode, button off, default).



## Cyclic mode

### Cyclic mode

Cyclic mode switches the multiparameter into so-called cyclic mode. If you have say 4 banks, called A, B, C and D, and gradually increase the multiparameter value, it starts with A, then interpolates to B, then to C and finally to D. But after that you cannot interpolate back to A, because D is the last one, the maximum value. In cyclic mode the multiparameter behaves as if there were a clone of A at the end, hence after D is reached, the multiparameter interpolates back to A and creates a full circle A->B->C->D->A. This is handy for example if you use a saw wave modulator to drive the multiparameter and want to repeat the sequence of the banks.

## Interpolate values

### Interpolate values

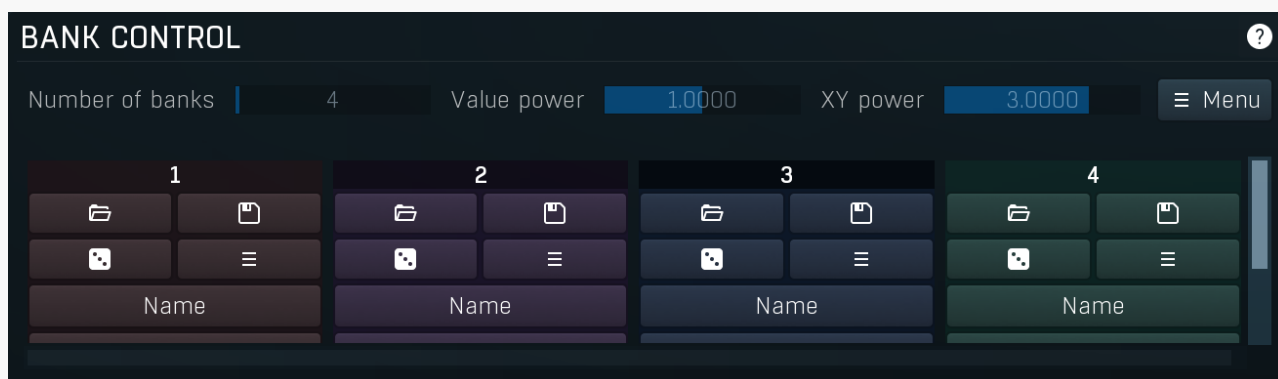
Interpolate values controls if the parameter value is to be interpolated between the bank values or if it will take the value from the nearest bank. For example, when bank A contains the value 0% for the parameter and bank B contains 100% and you set the multiparameter to 30%, then when interpolation is enabled, 30% is selected for that parameter, when the interpolation is disabled, the nearest value, 0%, is selected. If you want the parameter to step from one bank value to another then disable interpolate values.

## Set interpolate to all parameters

### Set interpolate to all parameters buttons

Set interpolate to all parameters buttons sets the interpolate values setting for all parameters controlled by that multiparameter.

# Bank control panel



Bank control panel is available only in **Banks mode** and contains tools to define the banks between which the multiparameter is interpolating. The multiparameter stores parameter values for each bank. Here you can load and save these values. Each bank has 5 buttons and a value for each controlled parameter. Click the **load button** to load the bank values into the plug-in. If you want to change say bank 3, you first click its **load button**, change whatever you need and resave the settings. By clicking the **save button** you overwrite the bank's settings from those currently set in the plug-in. A typical approach to define the multiparameter's behaviour is to set the number of banks, then go to the plugin editor, set all associated parameters to the values you would like to have in bank 1 and click the save button for bank 1, then modify the parameters to whatever you want in bank 2 and click the save button for bank 2, etc. You can also use the **Random button** to generate random values using the smart-randomization engine for each of the banks. And the **menu button** enables you to re-order the banks

For each bank, the values for each parameter are shown and can be changed as desired.

## Number of banks

4

### Number of banks

Number of banks controls the number of settings that the multiparameter stores for all parameters. By changing the multiparameter value all associated parameters are then modified according to these settings. Please note that when you change the number of banks, the multiparameter will behave differently, because the multiparameter's range from 0% to 100% will now be distributed between a different number of presets. If you had automated the multiparameter value in your host for example you will almost certainly need to edit / rewrite the automation envelope.

## Value power

1.0000

### Value power

Value power lets you post-process values for each bank by specific power function. This either leads to higher values for low powers, or lower values for high powers. The same thing can be implemented using transforms, but this is much easier considering the potential number of parameters you may have there. It is especially useful for circular XY pads. Use trial-and-error approach to set this up to your liking.

## XY power

3.0000

### XY power

XY power is used only if **XY-pad between banks** is enabled and controls how the engine produces values for individual parameters. The higher the number is, the more will the engine separate them. Use trial-and-error approach to set this up to your liking.

## Menu

### Menu

Menu button provides some additional features for processing the entire set of banks.

**Sort banks (up)** reorders the banks so that the values of the selected parameter are in increasing order.

**Sort banks (down)** reorders the banks so that the values of the selected parameter are in decreasing order.

**Reverse** reverses the order of banks, so that the first bank contains values of the previously last one and so on.



**Interpolate** lets you change the number of banks, but keeps the values as they are now by calculating values of parameter for all banks. It is usually useful when you want to provide 'banks in between current banks', without manually calculating the new values. **Auto-gain** (if available) temporarily enables AGC and automatically sets up the main plugin gain to each bank so that all banks provide similar output loudness. To use it, ensure that the main gain parameter is attached to the multiparameter, start playback of your sound material and press this button. It will take several seconds to complete depending on the number of the banks. **Set names by values** sets the names for each bank to the values of the selected parameter. It may be handy when replicating existing parameters for example.



### Load

Load button loads the bank settings by setting all associated parameters to the values in the particular bank.



### Save

Save button saves the current values of all associated parameters into the particular bank. So you can edit all those parameters in the plugin then click the save button to store them in the bank.



### Randomize

Randomize button loads random settings to the bank using the smart randomization engine. Only parameters associated with the multiparameter are randomized.

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.

Hold **Shift** while clicking the button to undo the previous randomization.



### Menu

Menu button provides some additional options related to the bank.

Name

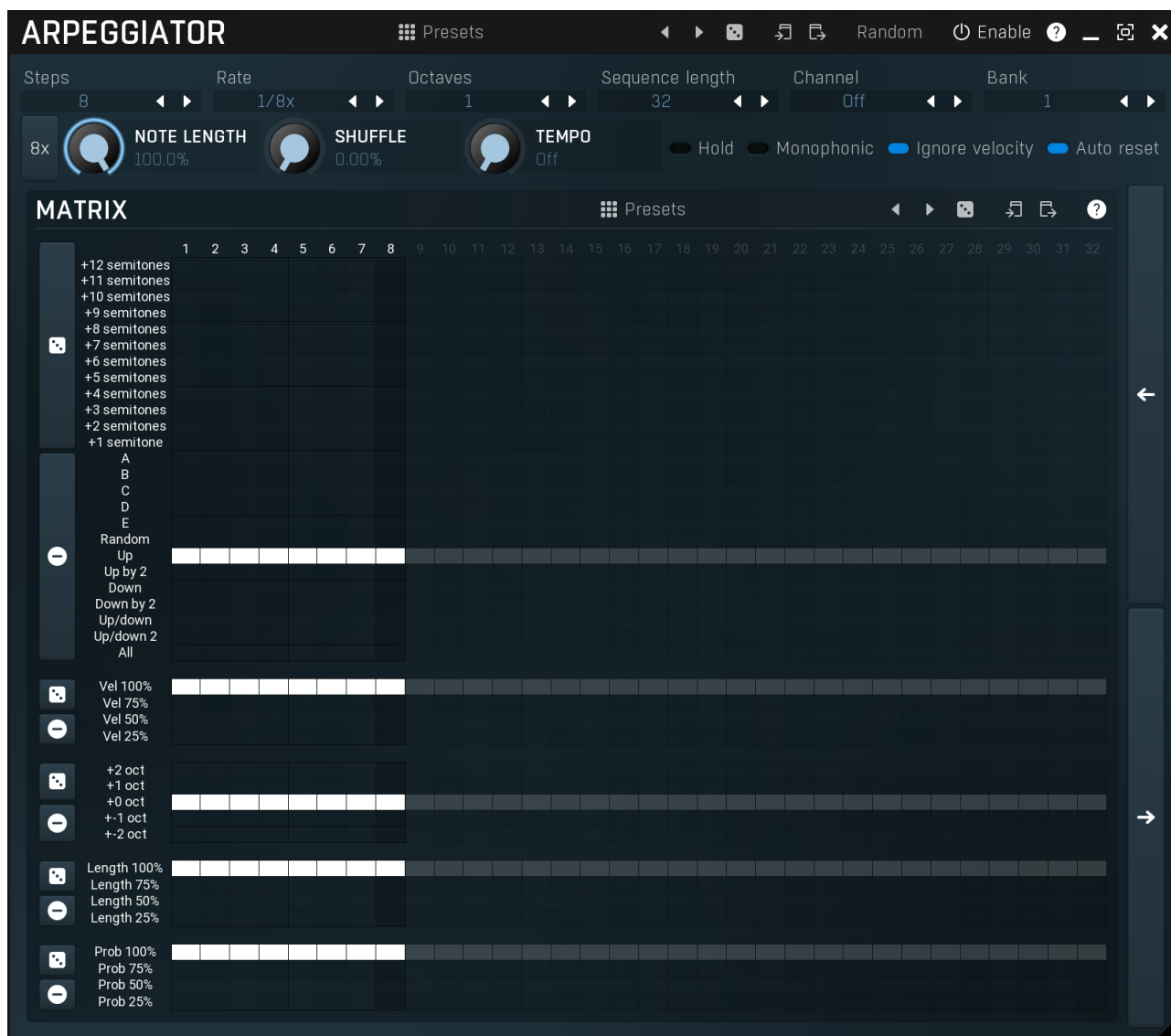
### Name

Name button lets you rename the bank.

### Name check

Name check button lets you rename the bank. This is a secondary name used for controls such as checkboxes and selectors if defined.

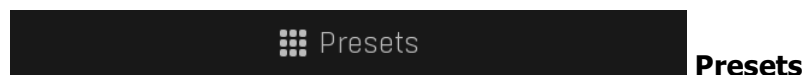
# Arpeggiator editor



Arpeggiator is a powerful sequencer which transforms the notes that you are playing into potentially long evolving sequences. The idea came from guitar arpeggios, where the left hand is holding the strings, defining the notes to be played, but the right hand is actually playing them, in any order.

This arpeggiator is based on the sequencer matrix, which enables generators in each step. You can imagine a generator is much like the guitarist's finger, but the generator is actually able to play different notes, in fact each generator can play a different note in each step. Now imagine a simple sequence - an empty matrix with just one row and 3 steps, say **Up**, enabled on each step. This would mean that on those steps the **Up** generator is activated and will emit exactly one note, just as a guitarist's finger hitting the string once. And the **Up** generator is playing one note from the chord you are holding, in upwards order. Hence if you hold say C major (C, E, G), the generator is playing C, E, G, C, E, G, ... And of course you may not be holding a chord, but say just a single note, say C. Well, then the sequencer is just repeating C. of course, the sequence can switch between multiple octaves, say, C0, C1, C2, C0..., depending on the **Octaves** control. Each of the generators works separately and all of them let you create complicated evolving sequences, which may eventually can generate sequences much longer than the 32 steps. In the example above the simple C major chord contains 3 notes, spread between say 4 octaves gets you 12 notes, and you may enable the **Up** generator in just a single step. That way the generator is triggered just once every sequence and each time it chooses the next note. If you enable all the steps then it will take 32 (steps) \* 3 (notes) \* 4 (octaves) = 384 steps before the sequence starts repeating. If you instead use both **Up/Down** and **Up 2/Down 2** generators, again enabled in just a single step, C major chord, 4 octaves, (I'll spare you the boring mathematics) it will take 4,224 steps before the sequence starts repeating. In the default 1/16x synchronization and 120bpm it makes almost 9 minutes! So the arpeggiator lets you build a fully non-repetitive 9 minute song just by pressing 3 tones!

And it can get much more complicated...



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

Random button generates random settings using the existing presets.

Steps

8



## Step count

Step count controls the number of steps in the arpeggiator sequencer. Please note that **up/down** and similar generators keep their own order of notes, so for example if you enable **up** for all steps, then it doesn't really matter how many steps you use.

Range: 1 to 32, default 8

Rate

1/8x



## Rate

Rate defines the sequencer speed. It is always in sync with the host and this parameter essentially controls the speed. The default value of 1/16 means each step takes 1/16th. So in monophonic mode there would be 1 note emitted every 1/16th, 4 times per quarter note.

Octaves

1



## Octaves

Octaves controls how many octaves the arpeggiator can produce. 1 octave means only the notes you are holding can be generated. A higher number makes it possible to emit the same notes in higher octaves.

Range: 1 to 5, default 1

Sequence length

32



## Sequencer length

Sequencer length controls the maximum sequence length for **up, down...** and other generators. It lets you provide some sense of rhythm. *For example, if you hold 4 notes and the number of octaves is 3, then 12 notes are generated by say **up** generator before the sequence starts from the beginning. You may want to select a sequencer length of 8, so that it starts repeating every 8 notes, which may be more natural in a typical 4/4 rhythm.*

Range: 1 to 32, default 32

Channel

Off



## Channel

Channel controls the MIDI channel the arpeggiator produces notes on. Of the value is not 'off', being the minimum, the arp will pass the input notes through and produce the arpeggiator notes on the selected channel. This can be used together with MIDI filtering to make different modules respond to different notes - normal and arp notes.

Range: Off to 16, default Off

Bank

1



## Bank

Bank lets you create multiple arpeggiator matrices and switch between using automation, modulation etc.

Range: 1 to 64, default 1

8x

## 8x

8x button increases the note length to be 8x longer. This way the notes may start to overlap. It is useful for arpeggiated pads for example. Please note that because the notes start overlapping, the CPU consumption may rise significantly for some sounds.



NOTE LENGTH

100.0%

## Note length

Note length controls the length of each note. 100% basically produces legato as each note is directly followed by the next one. Lower values makes the emitted notes shorter, leading to staccato.

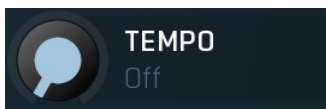
Range: 0.00% to 100.0%, default 100.0%



## Shuffle

Shuffle modifies the timing so that the arpeggiator stops producing straight notes, but rather shuffle (for a value of 100%) or even dotted (for a value of 200%).

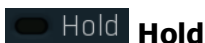
Range: 0.00% to 200.0%, default 0.00%



## Tempo

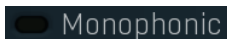
Tempo controls the tempo, which is normally synced to host, but you can unsync the arp and let it flow on its own using this control.

Range: Off to 3200.0000 bpm, default Off



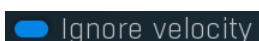
## Hold

Hold switch makes the arp listen only to note-on messages - pressing a key once enables it and pressing it second time disables it.



## Monophonic

Monophonic switch enables or disables the monophonic mode. This doesn't do anything on its own and affects only the editor behaviour. When enabled, any change in the sequence will ensure that the modified steps contain only one note (generator). If disabled, each step may contain multiple notes (generators) and the output would be polyphonic.



## Ignore velocity

Ignore velocity switch lets the arpeggiator ignore the velocity of the notes that you play. If enabled, the emitted note velocity is defined solely by the velocity matrix. If disabled, the velocity matrix still stands, but the velocity of the input notes still matters.



## Auto-reset

Auto-reset switch makes the arpeggiator reset the synchronization every time a note is pressed.

# Sequencer matrix

**MATRIX** Presets

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
+12 semitones																																
+11 semitones																																
+10 semitones																																
+9 semitones																																
+8 semitones																																
+7 semitones																																
+6 semitones																																
+5 semitones																																
+4 semitones																																
+3 semitones																																
+2 semitones																																
+1 semitone																																
A																																
B																																
C																																
D																																
E																																
Random																																
Up																																
Up by 2																																
Down																																
Down by 2																																
Up/down																																
Up/down 2																																
All																																
Vel 100%																																
Vel 75%																																
Vel 50%																																
Vel 25%																																
+2 oct																																
+1 oct																																
+0 oct																																
+1 oct																																
+2 oct																																
Length 100%																																
Length 75%																																
Length 50%																																
Length 25%																																
Prob 100%																																
Prob 75%																																
Prob 50%																																
Prob 25%																																

Sequencer matrix controls which notes are emitted in each step. Each step is defined in one column and each row enables a particular generator in each step. These generators emit exactly one note in each step; the difference is which note is emitted. In polyphonic

mode you can enable more than one generator in each step.

**A, B, C, D, E** generators emit the notes you are playing. A is the lowest note, B the one above it etc. Therefore enabling any of these generators will always produce the same note. You can therefore sequence 5 notes freely. If you are holding less than 5 notes, say 3, then D is the same as A and E is the same as B. These generators ignore settings of the **Octaves** parameter, so the played notes only are emitted.

**+n semitones** generators emit the note transposed upwards from the note A. For example if you are holding C major chord (C, E, G), everything is related to C being the lowest note. The +1 semitone generator then produces C#, +2 semitones is D etc. Therefore enabling any of these generators will always produce the same note. If you have selected A, B, C and +9 semitones in 4 steps, you will hear C, E, G and A (which is 9 semitones up from C). These generators ignore settings of the **Octaves** parameter, so the played notes only are emitted. They are displayed above the **A, B, C, D, E** generators to give a good visual representation of the transpositions.

**Up** generator produces sequence starting from the lowest note up, one by one, including higher octaves. For example, if you are holding C major chord (C0, E0, G0) and the **Octaves** is set to 2, then the generator produces the sequence: C0, E0, G0, C1, E1, G1, C0 ... **Up by 2** is similar, but steps by 2, hence the resulting sequence is: C0, G0, E1, E0, C1, G1, C0 ... **Down** generator is the exact opposite of Up and the resulting sequence is: G1, E1, C1, G0, E0, C0, G1 ... **Down by 2** generator is similar, but again steps by 2, hence the resulting sequence is: G1, C1, E0, E1, G0, C0, G1 ...

**Up/down** generator is a basically a merge of Up and Down generators, going from the lowest note up and then back down, so in the example above the result is: C0, E0, G0, C1, E1, G1, E1, C1, G0, E0, C0... **Up/down 2** generator is similar, but repeats the limit notes, so the sequence is: C0, E0, G0, C1, E1, G1, G1, E1, C1, G0, E0, C0, C0... Combining both Up/down and Up/down 2 generators can produce various evolving sequences.

**Random** generator chooses a random note every time.

**All** generator plays all of the notes every time. It is useful for trance-gate-like effects.

 Presets

## 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.



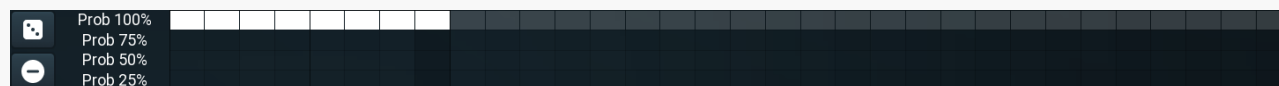
### Copy

Copy button copies the settings onto the system clipboard.



### Paste

Paste button loads the settings from the system clipboard.



## Probability matrix

Probability matrix lets you adjust the probability that a note in the step will be emitted. The setting affects all notes in the step, so if you are using polyphonic mode, it is not possible to let different notes in a single step have a different probability.



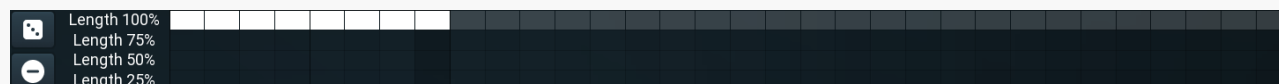
### Randomize

Randomize button generates a random probability matrix.



### Clear

Clear button sets the probability matrix to its defaults.



## Length matrix

Length matrix lets you adjust the length of each note. This setting works in conjunction with the global **Length** parameter. The setting affects all notes in the step, so if you are using polyphonic mode, it is not possible for different notes in a single step to have different lengths.



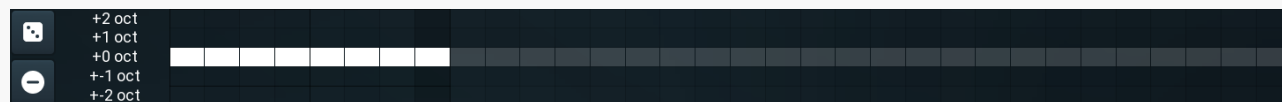
### Randomize

Randomize button generates a random length matrix.



### Clear

Clear button sets the length matrix to its defaults.



## Octave matrix

Octave matrix lets you transpose all of the notes in each step by a certain number of octaves. The setting affects all notes in the step, so if you are using polyphonic mode, it is not possible to let different notes in a single step be transposed by different number of octaves.



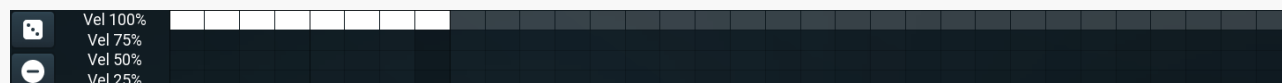
### Randomize

Randomize button generates a random octave matrix.



### Clear

Clear button sets the octave matrix to its defaults.



## Velocity matrix

Velocity matrix controls the velocity of the notes in each step. The setting affects all notes in the step, so if you are using polyphonic mode, it is not possible to let different notes in a single step have a different velocity.

If **Ignore velocity** is enabled, then the actual velocities of the notes you are holding do not matter and the values in the velocity matrix control the actual velocities of the output notes. These are defined as percentages, but if you are used to the ancient MIDI standard, which uses velocities 0-127, then 100% equals 127, 50% is 63 (or 64) etc.

If **Ignore velocity** is disabled, then the actual velocities are affected by both the velocity of the notes you are holding and the **Velocity matrix**. For example, if you are holding a note with velocity of 60 and the value in the matrix is 50% then the output note velocity will be 30.



### Randomize

Randomize button generates a random velocity matrix.



### Clear

Clear button sets the velocity matrix to its defaults.



### Randomize

Randomize button generates a random sequencer matrix.



### Clear

Clear button sets the sequencer matrix to its defaults.



### **Left**

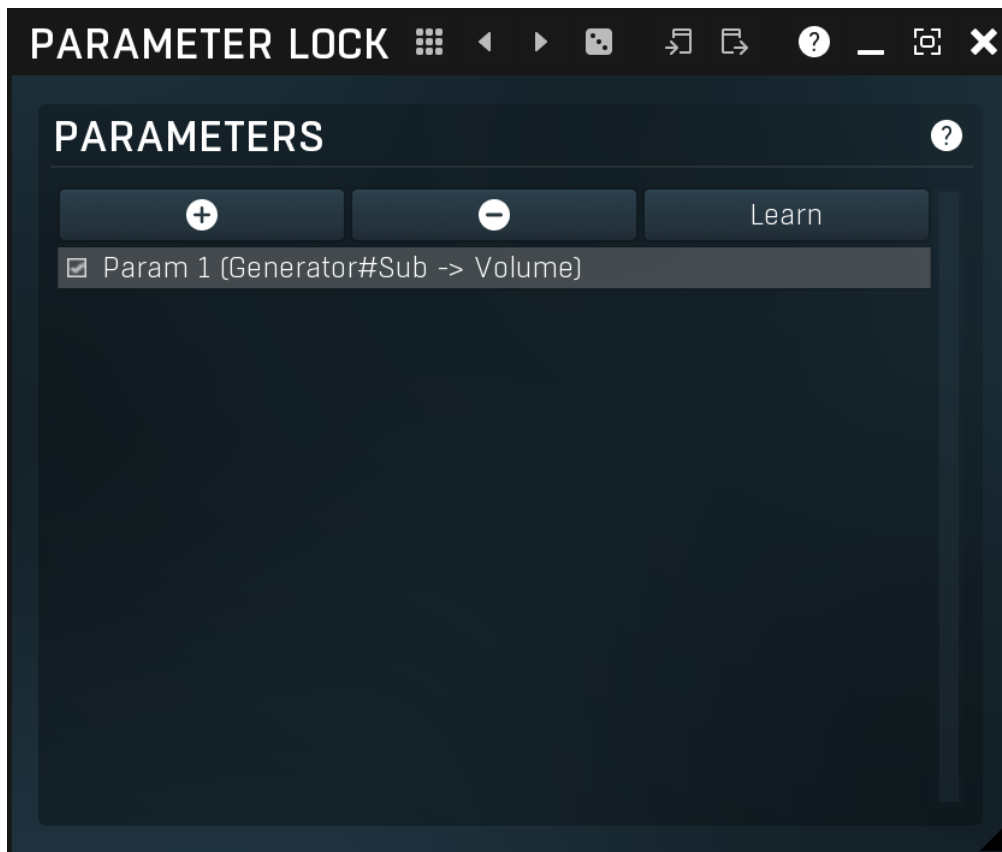
Left button shifts the whole sequence one step to the left.



### **Right**

Right button shifts the whole sequence one step to the right.

# Parameter lock editor



Lock provides a simple way to keep some parameters unchanged when using randomization or browsing presets. You can still change these locked parameters by adjusting the control directly. You simply use the learn feature (right click) in the same way you would with modulators or multiparameters, and touch every parameter you want to keep locked. You can also select them directly in the Parameter Lock window where you can also save them as presets, copy & paste etc. Learning mode is ended by clicking the button again. Please note that this list is not saved with global plugin presets for obvious reasons. The parameters can be locked or unlocked directly in the list or by clicking the lock button associated with the parameter on the Easy screen.



## 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.

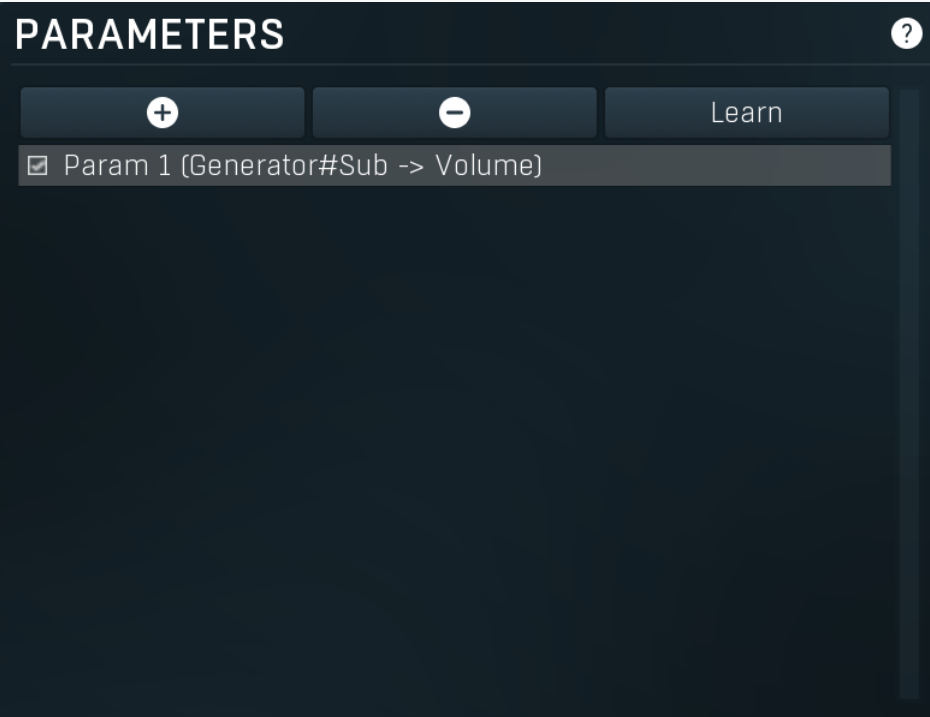


## Paste

Paste button loads the settings from the system clipboard.

## Parameters panel





Parameters panel configures the list of the parameters which are locked.



### **Add**

Add button adds a parameter to the list of locked parameters. Alternatively you can use the learn feature available by right-clicking the paramlock button for example.



### **Delete**

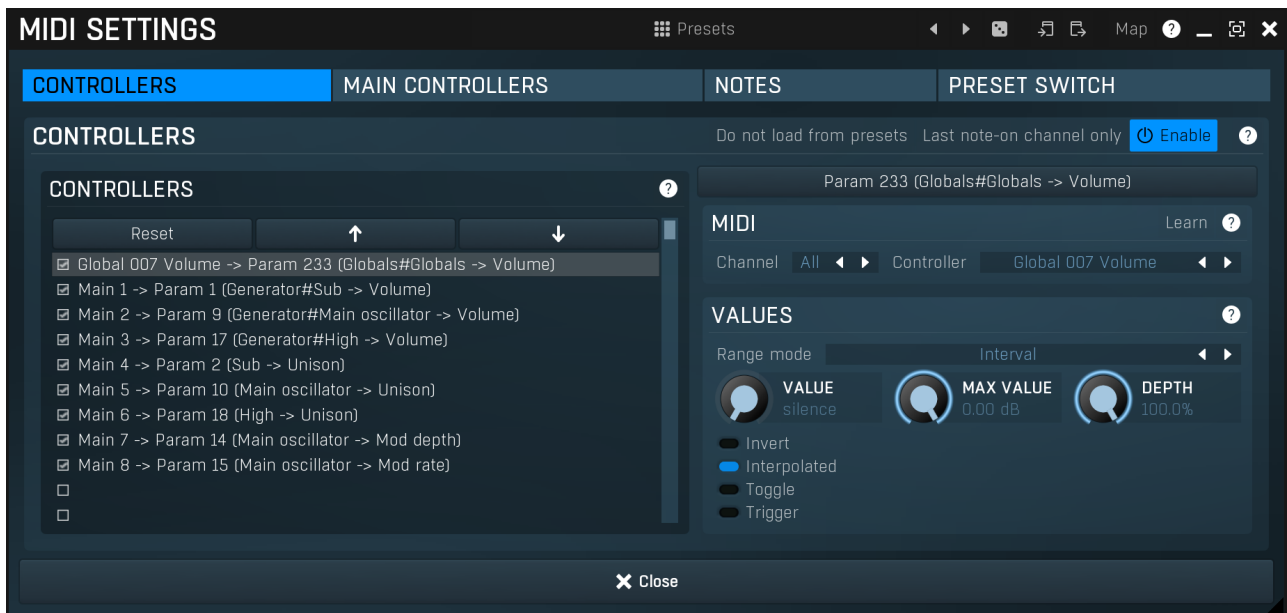
Delete button deletes the selected parameter from the list of controlled parameters.



### **Learn**

Learn button starts or stops the learning. Click it, then move some parameters in the plugin, then click it again. Learning can also be accessed from the global parameter lock menu.

# MIDI editor



MIDI settings window lets you configure, how the plugin reacts to various MIDI messages. You can use MIDI controllers or MIDI notes and you can also configure a controller to switch between presets, which is especially useful for realtime performances.

Presets

## 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.



### Map

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

CONTROLLERS

MAIN CONTROLLERS

NOTES

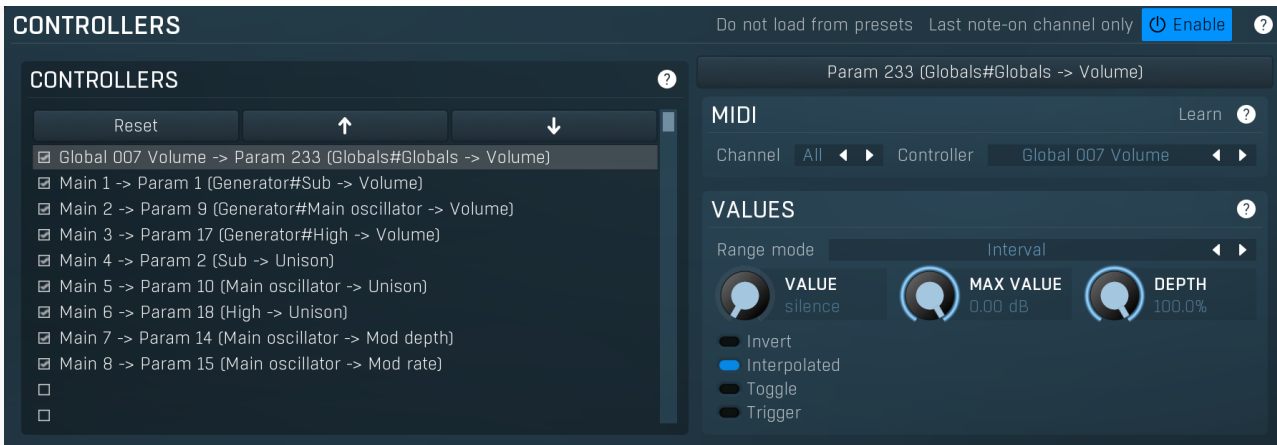
PRESET SWITCH

Tab

## selector

Tab selector switches between subsections.

## Controllers panel



Controllers panel contains settings of MIDI controllers.

### Do not load from presets

### Do not load from presets

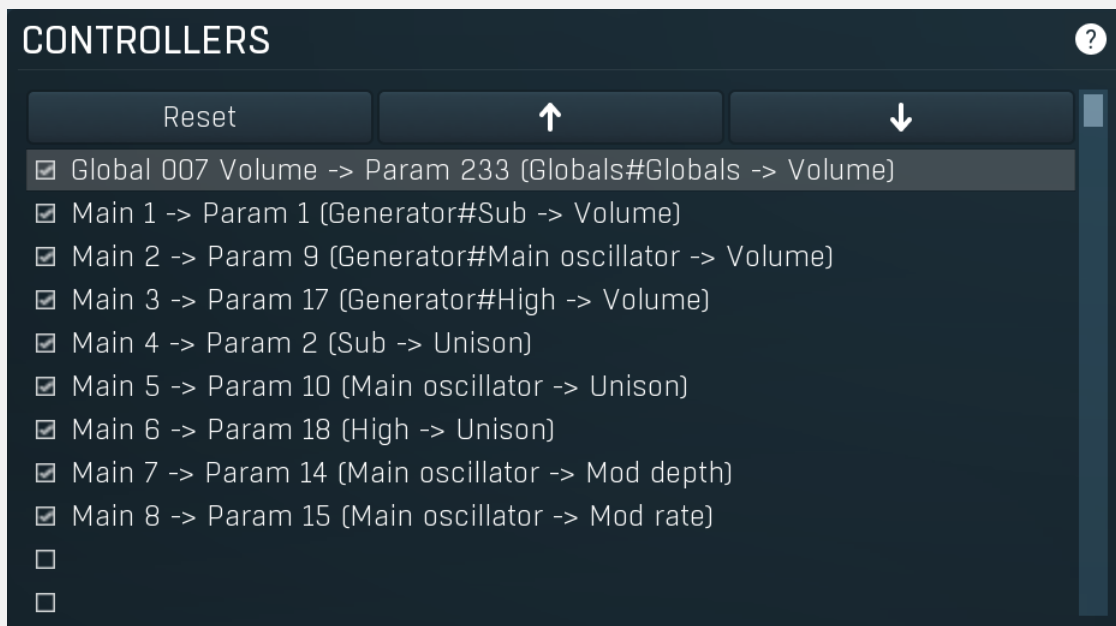
Do not load from presets button disables loading the controllers from presets. This may be handy if you have configured specific MIDI controllers with target parameters and you want to browse the presets without the need to configure them every time. Please note that some presets may rely on specific controllers though. For example, if a preset requires a velocity controller to provide velocity-dependent response, this option will avoid loading it, so the preset won't be complete, until you reconfigure it.

### Last note-on channel only

### Last note-on channel only

Last note-on channel only button makes the engine more suitable for voice-per-channel devices. These devices are able to send different controllers for each note you press, which however means that these could collide. This option makes the engine pass only the controllers that are related to the last note you pressed. For classic keyboards it is not relevant as you will usually use a single MIDI channel to transmit both the controllers and notes. Some more modern keyboard controllers will allow you to select one MIDI channel for the notes and a different one (or the same one) for the controllers.

## Controllers



Reset

### Reset

Reset button resets the selected controller to undefined state.



### Up

Up button moves the selected controller up one item, if possible. This may be useful when keeping things organized, but please note that if you have some multiparameter, modulator or another subsystem access the ranges of individual controllers, this function will reorder them, so these connections will no longer be correct.



### Down

Down button moves the selected controller down one item, if possible. This may be useful when keeping things organized, but please note that if you have some multiparameter, modulator or another subsystem access the ranges of individual controllers, this

function will reorder them, so these connections will no longer be correct.

Param 233 [Globals#Globals -> Volume]

## ParameterIndex

ParameterIndex button lets you choose the parameter being controlled. The set contains all automatable parameters.

## MIDI

### MIDI

Learn ?

Channel All ◀ ▶ Controller Global 007 Volume ◀ ▶

Learn

### Learn

Learn enables or disables MIDI learn. When enabled, the plugin listens to both the MIDI CC messages from the controllers that you touch and the target parameters that you touch and associates the last-touched of each with the selected slot. You can perform several mappings by selecting another slot, adjusting a hardware controller. Then adjusting a target parameter, and repeating those steps for each mapping desired.

Channel All ◀ ▶ **Channel**

Channel defines the controller MIDI channel.

Controller Global 007 Volume ◀ ▶ **Controller**

Controller defines the source controller.

## Values

### VALUES

?

Range mode Interval ◀ ▶



**VALUE**  
silence



**MAX VALUE**  
0.00 dB



**DEPTH**  
100.0%

- ☐ Invert
- ☒ Interpolated
- ☐ Toggle
- ☐ Trigger

Range mode Interval ◀ ▶ **Range mode**

Range mode defines how the parameter range is selected. While sometimes it is better to specify minimum and maximum, other times it is better to use a nominal center and depth (% of full scale). This control allows you to define which one it will be.

**Up and down** mode makes the values go above and below the selected **Value**, which is considered the center. The interval is made smaller if necessary.

**Full range mode** is similar, except the range is symmetrically constrained, so the selected **Value** may not be the center anymore.

**Up/down only modes** goes from the selected value up/down only.

Let's compare these 4 modes. Taking a value of -12dB value, with a depth of 75% and a scale of +/- 24dB. The nominal range is therefore = +/-24 dB \* 75% = 36dB. With values of 0%, 50% and 100% the outputs are:

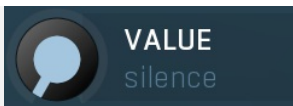
Up and down: -24, -12, 0 (range constrained to 12 dB either side)

Full range: -24, -6, 12 (range limited to minimum, but not constrained)

Up only: -12, 6, 24 (range not constrained = +/-24 dB \* 75% = 36dB)

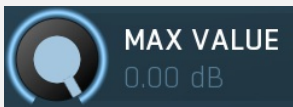
Down only: -12, -18, -24 (range limited to minimum)

**Interval mode** is the most simple one and goes from **Value** to **Maximal value**.



### Value

Value defines the center of the target parameter's range or the minimum if the **Range mode** is set to **Interval**.



### Maximal value

Maximal value defines the upper limit of the target parameter's range. It is available only if the **Range mode** is set to **Interval**. This value can be lower than **Value**. 0% is always mapped to reference>Value and 100% to reference>Maximal value.



### Depth

Depth defines size of the target parameter's range. It is used only if the **Range mode** is not set to **Interval**.



### Invert

Invert checkbox inverts the controller shape, so the minimum becomes the maximum etc.



### Interpolated

Interpolated makes the controller value interpolated over the time using the smart interpolation. This approach ensures there won't be abrupt changes, which could lead to clicks and pops. However sometimes you may want to apply these changes immediately - for example when changing ADSR based on the note velocity, in which case this parameter should be disabled.



### Toggle

Toggle mode makes the controller switch between the maximum and minimum of the target parameter whenever triggered. By default triggering it means going from values below 50% to above 50%. By enabling **Trigger** you can make it perform the trigger everytime the value is changed.



### Trigger

Trigger mode makes the controller automatically produce maximum and the minimum right after it. It can be handy with some buggy MIDI controllers providing buttons, which however do not send value 0, and only repeat value 127. Trigger makes it behave like the minimum was actually sent by the MIDI controller a little bit after the original message.

## Main controllers panel



Main controllers panel lets you define the set of main MIDI controllers on your MIDI device. These are not stored with the presets, so using them lets you easily switch between MIDI controllers, create presets that will work for users of other MIDI controllers etc. Using the Main controllers is no different than using the standard MIDI controllers, but the extra 'layer' can make things simple when using multiple controllers and also in general situations where your MIDI device has several controllers with quite 'random' numbers.



### 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.

Global 001 Modulation wheel



### Controller

Controller defines the MIDI controller associated to this Main controller.



### MPE

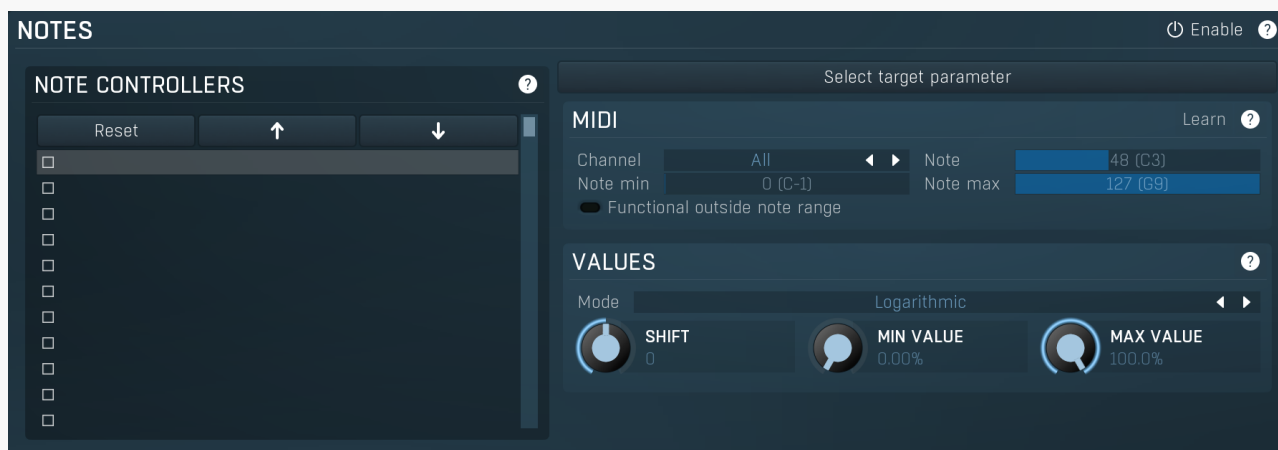
MPE switch makes the associated MIDI controller MPE compatible meaning that it will recognize different values for each MIDI channel and that can be exploited using per-voice modulation. Use this option **ONLY** for custom MPE controllers, which have further per-voice controllers available.

Learn

### Learn

Learn enables or disables MIDI learn. When enabled, the plugin listens to the controllers you touch and associates them to the main controller.

## Notes panel



Notes panel contains settings of MIDI note controllers, if you want to control parameters using MIDI keys.

## Note controllers

## NOTE CONTROLLERS



Reset



Reset

### Reset

Reset button resets the selected controller to undefined state.



### Up

Up button moves the selected controller up one item, if possible. This may be useful when keeping things organized, but please note that if you have some multiparameter, modulator or another subsystem access the ranges of individual controllers, this function will reorder them, so these connections will no longer be correct.



### Down

Down button moves the selected controller down one item, if possible. This may be useful when keeping things organized, but please note that if you have some multiparameter, modulator or another subsystem access the ranges of individual controllers, this function will reorder them, so these connections will no longer be correct.

Select target parameter

### Learn

Learn enables or disables MIDI learn. When enabled, the plugin listens to both the notes you touch and the parameters you touch and associates them with the selected slot.

## MIDI

### MIDI

Learn ?

Channel All ◀ ▶

Note 48 [C3]

Note min 0 [C-1]

Note max 127 [G9]

Functional outside note range

Channel All ◀ ▶ Channel

Channel defines the controller MIDI channel.

Note 48 [C3] Note

Note defines the controller's target MIDI note. It is used only in On/off and Switch modes, which you can set using **Mode** parameter (in the **Values** panel).

Note min 0 [C-1] Note min

Note min controls the lowest note to be used by a controller in Linear or Logarithmic mode. The minimum value of the target parameter will then be associated to this note.

If both Note min and Note max parameters are default, the plugin takes the actual frequency of each note, and transforms it into the range 20Hz to 20kHz, which is the range used by all equalizers and filters, so that you can literally play a parameter on a MIDI keyboard. If you change either of these 2 parameters however, the plugin takes the range of notes as the requested interval. This is useful for example if you have a small MIDI keyboard used for soloing and you want increase some parameter the higher you play. In the default mode it would be difficult, since the range of frequencies is much bigger than the range of your MIDI keyboard. Set the **Note min** and **Note max** to C0 and B0 respectively, the **Mode** to Logarithmic and select a suitable target parameter (Dry/Wet

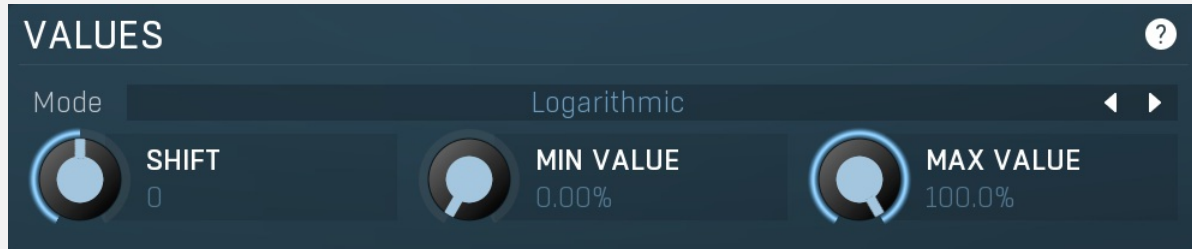
is fine). Send MIDI notes in the specified range to the plugin and you will see the target parameter increase (by 9.09% (= 100 / (12-1)) for a 100% range).

Note max 127 (G9)

### Functional outside note range

Functional outside note range makes the note controller work even if the note isn't in the specified range, clamping the value to the minimum or maximum.

## Values



Mode Logarithmic Mode

Mode controls how the controller works.

**Key** takes the note index and transforms it into 0..1, which is the output of any controller. This mode is useful for scale switches for example - if you want to use MIDI keys to change values linearly.

**Key (in octave)** is similar but it has only 12 values - one per each key and it doesn't matter which octave you press it in.

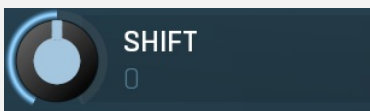
**Linear** converts the notes into frequencies and then transform them into the linear scale from 20Hz to 20kHz.

**Logarithmic** converts the notes into the frequencies and then into the logarithmic scale from log(20) to log(20000). A typical use case is when you want to control an equalizer band using a MIDI keyboard. Since EQ frequencies work in logarithmic scale, this mode makes both things compatible and the EQ frequency will be set to the note frequency.

**On/off** modes react only to single notes and can be used for triggers. When the Note On is received the parameter is changed to its **Max value** and when the Note Off is received the parameter is changed to its **Min value**. So this mode can also be used to change between any 2 parameter values.

**Switch** modes are similar, but only recognize when a note is pressed. The Note Offs are ignored. Note Ons select the **Max value** and **Min value** alternately. In all octaves mode it doesn't matter which octave is used. For example, this is useful when you want to use any note C to switch something on and off.

**Velocity** modes do not actually follow the note number being pressed, but it's velocity instead. While you can do the same thing with normal MIDI controllers using the special Velocity controllers, this one allows you to select only some notes to follow.



### Shift

Shift lets you shift the original note up or down by the specified number of semitones.



### Min value

Min value defines the minimum value for the target parameter.



### Max value

Max value defines the maximum value for the target parameter.

Enable MIDI program change

Enable

### MIDI program change

Enable MIDI program change enables processing program change MIDI message.

Enable bank & program change for the categorizer

Enable



### bank & program change for the categorizer

Enable bank & program change for the categorizer activates browsing devices using the bank select MIDI messages. Program change MIDI message then switches between the device presets.

## Preset previous/next trigger panel



## PRESET PREVIOUS/NEXT TRIGGER

Learn  Enable 

Channel All  
Controller Global 000 Bank select

Preset previous/next trigger panel lets you select a MIDI controller, which will switch presets. It provides the same action as clicking the arrows next to the main preset button. When the controller value gets below 33%, the previous preset is loaded. When the controller value gets above 66%, the next preset is loaded.

Learn

### Learn

Learn enables or disables MIDI learn.

Channel All

### Channel

Channel defines the controller MIDI channel.



Controller Global 000 Bank select

### Controller

Controller defines the source controller.

## Simulate program change via controller panel

### SIMULATE PROGRAM CHANGE VIA CONTROLLER

Learn  Enable 

Channel All  
Controller Global 000 Bank select  
Number of values 128

Simulate program change via controller panel lets you select a MIDI controller, that will work as program change, for convenience. You can use it then to switch between A-H presets or presets via panel below.

Learn

### Learn

Learn enables or disables MIDI learn.

Channel All

### Channel

Channel defines the controller MIDI channel.

Controller Global 000 Bank select

### Controller

Controller defines the source controller.

Number of values 128

### Number of values

Number of values defines the number of programs to switch between. By default Program change MIDI standard offers 128 programs. However it may be too many and could be hard to actually control with the specific controller. Hence you can lower the number of actual programs.

## Program change in presets panel

### PROGRAM CHANGE IN PRESETS

 Enable 

Folder PROGRAMS  
Channel All

Program change in presets panel enables the MIDI program change processing. If disabled, the plugin follows Program Change messages by changing the A-H presets. The obvious disadvantage is that this way there are just 8 presets. By enabling this feature the plugin stops selecting A-H presets and rather loads different presets from the specified preset folder, including all sub-folders. The default folder is called "Programs". To use it, you simply need to create a preset folder called Programs and put the presets into it. Note that the order matters of course. And you can change the folder name at any time, so you can have several sets of selectable presets.

Folder PROGRAMS

### Folder

Folder defines the preset folder from which the presets for program-change MIDI messages are taken.

Channel

All



## Channel

Channel defines the program change MIDI channel.

# Used controls

Here we discuss the general properties of all application controls. As a most important rule you should note, that you can always use any question mark button or F1 (or Ctrl+F1 or Ctrl+H) key with the mouse cursor over a specified control to get detailed information about what it does and how to use it.

## Value button

A dark blue rectangular button with the text "Semitones" in white on the left and a white numerical value "0" on the right.

Value button is an alternative to the knobs and its main advantage is that it is very small. In some cases the button simply serves as a clickable item and a menu is shown when clicked. However the mouse wheel and other controls still do work.

- **Click and drag using the left mouse button** to change the value.
- **Right mouse button** shows a menu with additional options. Hold **ctrl/cmd** to select the default value.
- **Mouse wheel, arrow keys** and vertical drag using **middle mouse button** or using **left mouse button while holding Ctrl** modifies the value more precisely.
- **Home key** configures the minimal possible value, conversely **end key** setups the maximal one.
- **Esc or Backspace keys** restore the original value when either one is pressed during dragging.
- **Shift + left mouse button** or **double-click using left mouse button** lets you edit the value as text. You can use the virtual keyboard or type on your computer keyboard. In some cases this shows a menu with all possible values instead.
- **Alt + press then release** measures the time between the press and the release and applies it as time/frequency tap. Usable only for certain values of course.


## Switcher

A dark blue rectangular control with the text "Quality" in white on the left, "High" in light blue in the center, and two white arrow icons (left and right) on the right.

Switcher is an alternative to a tracker or knob control, but it has a limited set of values.

- **Left mouse button** shows a menu with list of all possible values. This function might be unavailable in certain cases when the number of possible values is too high.
- **Right mouse button** shows a menu with additional options. Hold **ctrl/cmd** to select the default value.
- **Up** and **Down** arrow keys, **buttons** in the control and **mouse-wheel** increase or decrease the value.

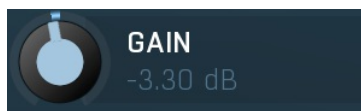
## Zoomer

A thin horizontal bar with a light blue gradient and a small icon on the right side.

Zoomer provides a simple way to zoom and move in an enlargeable view.

- **Plus button** zooms-in.
- **Minus button** zooms-out.
- **Zoom default button** zooms to the default ratio, which typically means full zoom-out.
- **Lock button** locks the zoom ratio.

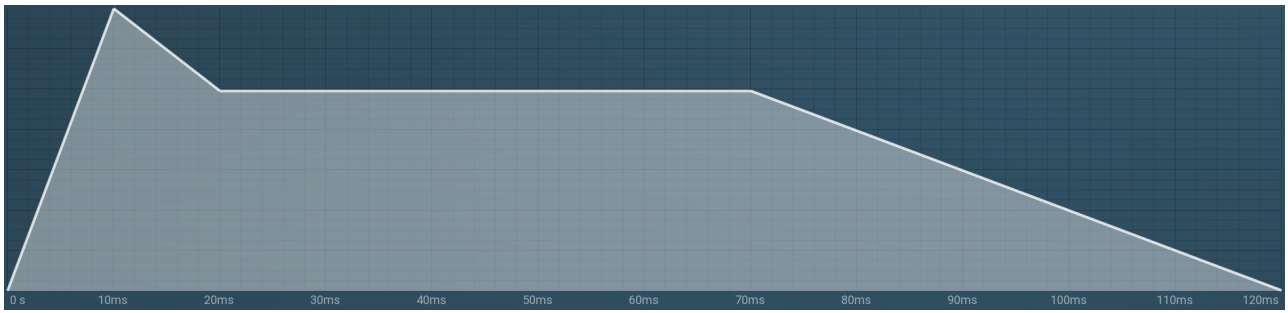
## Knob



Knob simulates physical knobs used to edit various values.

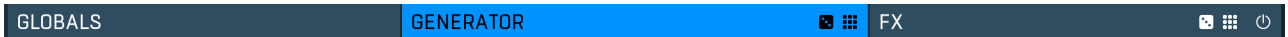
- **Click and drag using the left mouse button** to change the value.
- **Right mouse button** shows a menu with additional options. Hold **ctrl/cmd** to select the default value.
- **Mouse wheel, arrow keys** and vertical drag using **middle mouse button** or using **left mouse button while holding Ctrl** modifies the value more precisely.
- **Home key** configures the minimal possible value, conversely **end key** setups the maximal one.
- **Esc or Backspace keys** restore the original value when either one is pressed during dragging.
- **Shift + left mouse button** or **double-click using left mouse button** lets you edit the value as text. You can use the virtual keyboard or type on your computer keyboard. In some cases this shows a menu with all possible values instead.
- **Alt + press then release** measures the time between the press and the release and applies it as time/frequency tap. Usable only for certain values of course.

## Graph editor



Graph editor will show and edit one or more graphs.

## Tab-set



Tab-set is typically used wherever there is too much to edit, but not enough space to display it all. It can be also used to switch between possible alternatives.

- **Left mouse button** selects a tab.
- **Ctrl + Left mouse button or Right mouse button** displays the whole tab in a pop-up window (this is not used for all sets of tabs). This comes handy when you want to have multiple tabs visible at the same time.
- **Left and Right arrows** select the neighbouring tab.
- Click on one of the buttons on the border to scroll the control and show tabs that are currently invisible.

## Piano



Piano provides an easy way to mark a set of keys for some purpose.

- **Left mouse button** marks or unmarks a particular key.
- **Mouse-wheel** or **border buttons** scroll the piano if the display does not fit in the available space.

# Installation, activation, introduction to audio plugins

## Installation

All MeldaProduction plugins are currently available for Windows and Mac OS X operating systems, both 32-bit and 64-bit versions. You can download all software directly from our website. Since the installation procedures for the two operating systems are quite different, we will cover each one separately.

The download files for the effects include all the effects plug-ins and MPowerSynth. During the installation process you can select which plug-ins or bundles to install. If you have not licensed all of the plugins in a bundle then you just need to activate each plugin separately.

If you have multiple user accounts on your computer, always install the software under your own account! If you install it under one account and run it under a different one, it may not have access to all the resources (presets for example) or may not even be able to start.

## Installation on Windows

All plugins are available for VST, VST3 and AAX interfaces. The installer automatically installs both the 32-bit and 64-bit versions of the plugins.

**Note: Always use 32-bit plugins in 32-bit hosts, or 64-bit plugins in 64-bit hosts. 64-bit plugins cannot work in 32-bit hosts even if the operating system is 64-bit. Conversely, never use 32-bit plugins in 64-bit hosts. Otherwise they would have to be 'bridged' and, in some hosts, can become highly unstable.**

You can select the destination VST plugins paths on your system. The installer will try to detect your path, however you should check that the correct path has been selected and change it if necessary. In all cases it is highly recommended to use the current standard paths to avoid any installation issues:

32-bit Windows:

C:\Program files\VstPlugins

64-bit Windows:

C:\Program files (x86)\VstPlugins *(for 32-bit plugins)*

C:\Program files\VstPlugins *(for 64-bit plugins)*

If your host provides both VST and VST3 interfaces, VST3 is usually preferable. If a plugin cannot be opened in your host, ensure the plugin file exists in your VST plugin path and that if your host is 32-bit, the plugin is also 32-bit, and vice versa. If you experience any issues, contact our support via [info@meldaproduction.com](mailto:info@meldaproduction.com)

## Installation on Mac OS X

All plugins are available for VST, VST3, AU and AAX interfaces. Installers create both 32-bit and 64-bit versions of the plugins.

If your host provides multiple plugin interface options, VST3 is usually preferable. If you experience any issues, contact our support via [info@meldaproduction.com](mailto:info@meldaproduction.com)

Most major hosts such as Cubase or Logic should work without problems. In some other hosts the keyboard input may be partly non-functional. In that case you need to use the virtual keyboard available for every text input field. You may also experience various minor graphical glitches, especially during resizing plugin windows. This unfortunately cannot be avoided since it is caused by disorder in Mac OS X.

## Uninstallation on Windows

The Uninstaller is available from the Start menu and Control panel, in the same way as for other applications. If you don't have any of these for any reason, go to Program files / MeldaProduction / MAudioPlugins and run setup.exe.

## Uninstallation on OSX

The Uninstaller is available from Applications / MeldaProduction / MAudioPlugins / setup.app.

## Deleting all data, presets etc.

Even if you uninstall the plugins, some data will be left behind - because of potential crossdependencies or because these are your presets, settings, configurations etc. If you want to wipe out everything, please manually delete following folders:

Windows:

C:\ProgramData\MeldaProduction

C:\Users\{username}\AppData\Roaming\MeldaProduction

OSX:

Macintosh HD/Library/Application support/MeldaProduction/  
HOME/Library/Application support/MeldaProduction

## Performance precautions

In order to maximize performance of your computer and minimize CPU usage it is necessary to follow a few precautions. The most important thing is to keep your buffer sizes (latency) as high as possible. There is generally no reason to use latency under 256 samples for 44kHz sampling rates (hence 512 for 96kHz etc.). Increasing buffer sizes (hence also latency) highly decreases required CPU power. In rare cases increasing buffer sizes may actually increase CPU power, in which case you can assume your audio interface driver is malfunctioning.

You should also consider using only necessary features. Usually the most CPU demanding features are oversampling and modulation of certain parameters. You can reduce modulation CPU usage at the cost of lower audio quality in Settings/Settings/Modulator protection.

## Troubleshooting

The plugins are generally very stable, there are known problems however.

### GPU compatibility

The software uses hardware acceleration to move some of the processing (mainly GUI related) from your CPU (processor) to your GPU (graphics processing unit). It is highly recommended to use a new GPU, as it will provide higher performance improvements, and update your GPU drivers. Older GPUs are slower and may not even provide required features, so the software will have to perform all calculations in the main CPU. We also have had extremely bad experiences with GPUs from ATI and despite the fact that software is now probably bulletproof, it is recommended to use NVidia GPUs as there has not been a single case of a problem with them.

If you experience problems with your GPU (crashing, blank/dysfunctional GUI), and that you cannot disable the GPU acceleration from the plugin's Settings window itself, download this file:

<http://www.meldaproduction.com/download/GPU.zip>

And place the GPU.xml included in the zip into

Windows: C:\Users\{username}\AppData\Roaming\MeldaProduction  
Mac OS X: ~/Library/Application support/MeldaProduction

### Memory limits of 32-bit platform

Most hosts are now 64-bit ready, however some of them are not or users willingly choose 32-bit edition, because the required plugins are not 64-bit ready yet. All our software is 64-bit ready. Please note that you must NOT use the 64-bit plugins in 32-bit hosts, even if you have a bridge. If you are stuck with a 32-bit host for any reason, note that there is a memory limit (about 1.5 GB), which you may not exceed. This can happen if you load too many samples or different plugins for example. In that case the host may crash. There is no other solution than to use a 64-bit host.

## Updating

You can use "Home/Check for updates" feature in any of the plugins. This will check online if there is a newer version available and open the download page if necessary.

To install a newer (or even older) version you simply need to download the newest installer and use it. There is no need to uninstall the previous version, the installer will do that if necessary. You also do not need to worry about your presets when using the installer. Of course, frequent backup of your work is recommended as usual.

## Using touch-screen displays

Touch screen displays are supported on Windows 8 and newer and the GUI has been tweaked to provide a good workflow. Up to 16 connections/fingers/inputs are supported. Any input device such as touch-screens, mouse, tablets are supported. These are the main gestures used by the plugins:

- Tap = left click
- Double tap = double click
- Tap & hold and quickly tap next to it with another finger = right click. Tap & hold is a classic right-click gesture, however that doesn't provide a good workflow, so came up with this method, which is much faster and does not collide with functionality of some elements.

# Purchasing and activation

You can purchase the plugin from our website or any reseller, however purchasing directly from our website is always the quickest and simplest option. The software is available online only, purchasing is automatic, easy and instant. After the purchase you will immediately receive a keyfile via email. If you do not receive an e-mail within a few minutes after your purchase, firstly check your spam folder and if the email is not present there, contact our support team using [info@meldaproduction.com](mailto:info@meldaproduction.com) so we can send you the licence again.

To activate the software simply **drag & drop the licence file onto the plugin**. Unfortunately some hosts (especially on Mac OS X) either do not allow drag & drop, or make it just too clumsy, so you can use Home/Activate in any of the plugins and follow the instructions. For more information about activation please check the [online video tutorial](#).

You are allowed to use the software on all your machines, but only you are allowed to operate the software. The licences are "to-person" as defined in the licence terms, therefore you can use the software on all your computers, but you are the only person allowed to operate them. MeldaProduction can provide a specialized licence for facilities such as schools with different licence terms.

## Quick start with your host

In most cases your host will be able to recognize the plugin and be able to open it the same way as it opens other plugins. If it doesn't, ensure you did installation properly as described above and let your host rescan the plugins.

### Cubase

Click on an empty slot (in mixer or in track inserts for example) and a menu with available plugins will be displayed. VST2 version is located in MeldaProduction subfolder. However VST3 version is recommended and is located in the correct folder along with Cubase's factory plugins. For example, dynamic processors are available from the "Dynamics" subfolder.

To route an audio to the plugin's **side-chain** (if it has one), you need to use the VST3 version. Enable the side-chain using the arrow button in the Cubase's plugin window title. Then you can route any set of tracks into the plugin's side-chain either by selecting the plugin as the track output or using sends.

To route **MIDI** to the plugin, simply create a new MIDI track and select the plugin as its output.

### Logic

Choose an empty insert slot on one of your audio tracks (or instrument tracks for example) and select the plugin from the popup menu. You will find it in the Audio Units / MeldaProduction folder.

To route an audio to the plugin's **side-chain** (if it has one), a side-chain source should be available in the top of the plugin's window, so simply select the source track you want to send to the plugin's side-chain.

To route **MIDI** to the plugin, you need to create a new Instrument track, click on the instrument slot and select the plugin from AU MIDI-controlled Effects / MeldaProduction. The plugin will receive MIDI from that track. Then route the audio you want to process with the plugin to this track.

### Studio One

Find the plugin in the Effects list and drag & drop it onto the track you would like to insert the plugin to.

To route an audio track to the plugin's **side-chain** (if it has one), first enable the side-chain using the "Side-chain" button in the Studio One's plugin window title. Then you can route any set of tracks into the plugin's side-chain from the mixer.

To route **MIDI** to the plugin, simply create a new MIDI track and select the plugin as its output.

### Digital performer

In the Mixing Board, find an empty slot in the track you would like to insert the plugin to. Click on the field and select the plugin from the effects list.

To route an audio track to the plugin's **side-chain** (if it has one), choose the track you want to send using Side-chain menu, which appears at the top of the DP's plugin window.

To route **MIDI** to the plugin, simply create a new MIDI track in the Track view and select the plugin as its output.

### Reaper

Click on an empty slot in the mixer and a window with available plugins will be displayed. Select the plugin you want to open by double clicking on it or using Ok button.

*It is highly recommended to select all MeldaProduction plugins in the plugin window the first time you open it, click using your right mouse button and enable "Save minimal undo states". This will disable the problematic Undo feature, which could cause glitches whenever you change certain parameters.*

To route an audio track to the plugin's **side-chain** (if it has one), click on I/O button of the side-chain source track in the mixer. Routing window will appear, there you click "Add new send" and select the track the plugin is on. In the created send slot select the channels (after the "=>" mark) for the send, in stereo configuration 3/4 for example. Note that this way the whole track receives the side-chain signal and all plugins with it. It is possible to send it to a single plugin only, but it is more complicated, please check the Reaper's documentation about that.

To route **MIDI** to the plugin, create a new MIDI track and do the same thing as with side-chain, except you don't need to change output channels.

## Live

In Session view, select the track you would like to insert the plugin to. At the left top of Ableton Live's interface, click on the Plug-in Device Browser icon (third icon from the top). From the plug-ins list choose the plugin (from MeldaProduction folder), double click on it or drag & drop it into the track.

The X/Y grid usually doesn't provide any parameters of the plugin. This is because the plugins have too many of them, so you have to select them manually. Check Live's documentation for more information.

To route an audio to the plugin's **side-chain** (if it has one), select the track you want to send to the side-chain and in the 'Audio To' menu, choose the audio track that has the plugin on it. Then in the box just below that select the plugin from the menu.

NOTE: Live does NOT support any interface correctly, it doesn't use the reported buses properly, hence it doesn't work with surround capable plugins. Therefore you need to use VST version, which reports only stereo capabilities by default.

To route **MIDI** to the plugin, create a new MIDI track and in the 'MIDI to' menu, choose the audio track that has the plugin on it. Note that in Live only the first plug-in on any track can receive MIDI.

## ProTools

In the mixer click an empty slot to insert the plugin to and select the plugin from the tree. The plugin may be present multiple times, once for each channel configuration (mono->stereo etc.). As of now ProTools do not arrange them in the subfolders, which is a workflow dealbreaker, but we cannot do anything about it. The huge empty space on top of each plugin window, which occupies so much of the precious display area, is part of ProTools and every plugin window and again we cannot do anything about it. In some cases you may experience CPU overload messages, in which case please contact Avid for support. Note that ProTools 10 and newer is supported. RTAS compatibility for PT9 and older will never be added.

To route an audio to the plugin's **side-chain** (if it has one), open the plugin, click on the *No key input* button in the plugin title and select the bus you want the audio taken from. You might need to remember the bus number, unless your ProTools version supports bus renaming. ProTools doesn't support stereo (or surround) side-chains at all.

To route **MIDI** to the plugin, create a new MIDI track and in the mixer click the output field for that track and select the plugin, which should already be in the menu.

## FL Studio

First make sure plugins are scanned, either a full scan through the Plugin Manager or an automatic fast scan when you open the Plugin Database section of the browser in FL. The scanned plugins will show up in the Plugin Database > Installed section of the FL browser. The Effects and Generators sections in the Plugin Database will show all "favorite" plugins. These can be checked and unchecked in the Plugin Manager or added in some other ways. These favorites also show up in the Add menu, the menu for the "+" button in the channel rack, when you right click an existing channel button to replace or insert, in the plugin slot menu in the mixer and in the plugin picker (F8). The menus with favorite plugins also have a "More" choice that will show all scanned plugins. The full explanation is in our help file, on the page **Installing Plugins**.

To route an audio to the plugin's **side-chain**, first set up the mixer: make sure the track you want to receive audio from is sent to the track the plugin as a sidechain (**help**). Then set up the plugin wrapper: choose the desired input on the **Processing tab** of the wrapper options.

To route **MIDI notes** to the plugin, first configure the sender: choose a MIDI port for the input device in the MIDI settings (for a hardware device), or an output port in the **wrapper options** (for a VST plugin that produces MIDI). For the receiving plugin, set the input port in the wrapper options to the same value you chose in step 1.

To route **MIDI controllers**, the procedure is different. The usual method in FL is to link CC messages to plugin parameters (**help file**). VST plugins will also have 128 CC parameters published (through the wrapper) that can be linked this way. Those will send the specified CC MIDI message to the plugin, instead of changing a published parameter.

## GUI styles, editor modes and colors

MeldaProduction plugins provide a state of the art styling engine, which lets you change the appearance to your liking. The first time you run the plugins a style wizard will appear and let you choose the style and other settings. It may not be available in ProTools and other problematic hosts.

By default each plugin has a certain color scheme, which differs based on what kind of plugin is that. Also, sections of some plugins are colorized differently, again, based on what kind of section is that (this can be disabled in global settings). Despite you can change the colors anyhow you want, it is advantageous to keep the defaults as these are standardized and have predefined meaning, so just by looking at a plugin's color you can immediately say what kind of plugin and section is that. Same rules apply when designing devices for easy screens. This is the current set of colors:

Dynamics = orange



Equalization, filtering = green  
Reverb, delay = brown/yellow  
Modulation = blue  
Distortion, limiting = red  
Stereo = cyan/yellow  
Time, pitch, unison... = purple/pink  
Tools = grey

Special colors:  
Synchronization = grey  
Detection = blue/green  
Side-chain = green  
Effects = red  
Advanced stuff = grey



**MELDA** production  
*The only limit is your imagination*

## About MeldaProduction

The best sound on the market, incredible workflow and versatility beyond your imagination. We create the deepest and the most powerful audio plugins with unbelievable sound and tons of unique features you cannot find anywhere else.

## Innovative Thinking

At MeldaProduction, we make the most advanced tools for music production and audio processing. We get inspired by the whole range of tools from the ancient analog gear to the newest digital creations, but we always push forward. We've always felt the audio industry is extremely conservative, still relying on the prehistoric equipment making the job unnecessarily slow and complicated. That's why we invent new technologies, which make audio processing easier, faster, better sounding and more creative.

## Sound Matters

In the world full of audiophiles you just need superb audio quality. And that's why we spend so much time perfecting audio algorithms until they sound unbeatable. Everything from dynamic filters to spectral dynamic processing. Our technologies just sound perfect.

## Inspiring User Interface

Modern user interfaces must not only be easy and quick to use, but also versatile and the whole visual appearance should inspire you. MeldaProduction plugins provide the most advanced GUI engine on the market. It is still the first and only GUI engine, which is freely resizable and stylable. Our plugins can look as an ancient vintage gear, if you are working on old-school rock music. Or as super-modern

futuristic devices if you are working on modern electronic music.

## Easy to Use, Yet Versatile

The only limit is your imagination. Our plugins are with absolutely no doubt the most powerful and versatile tools on the market. Yet we managed to make the plugins easy to use via the devices and smart randomization system. But when you are ready, you are one click away from the endless potential the plugins provide.

## Never-Ending Improvements

Most companies create a plugin, sell it and abandon it. We improve our plugins, add features, optimize... until there is nothing left to improve and there are no more ideas. Unfortunately that hasn't happened yet :). And the best thing is that the updates are free-for-life!

*MeldaProduction was founded in 2009 by Vojtech Meluzin and is based in Prague, Czech Republic.*

**[www.meldaproduction.com](http://www.meldaproduction.com)**

**[info@meldaproduction.com](mailto:info@meldaproduction.com)**

MeldaProduction (c) 2017

