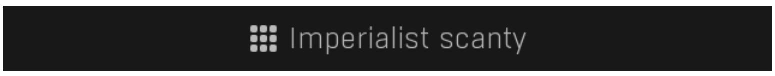
# MVocoder


MVocoder is an extremely advanced filter-based vocoder that you can use to generate a robotic voice, make your synth sing etc.

Vocoding in general is a voice-modulated synth sound. It works like this: the main input, called the **carrier**, is filtered into small spectrum regions called **bands**. You usually use a synth with some rich harmonic content for this input. The side-chain input, called the **modulator**, is then filtered the same way (or in a different way if you desire). You usually use a voice here. The modulator bands are then processed through an envelope follower, such that the levels of the modulator bands are measured. The levels are then applied to the carrier bands, which are combined to produce the output. *Simply put, only frequencies which exist in the modulator are preserved in the carrier.*

The technology was originally invented for compression in telephones, to save bandwidth and allow more calls to be carried simultaneously, hence the human voice has been the main interest. However it was soon adopted creatively for music, where you can generally use it for any type of signal. Please note however, that the carrier should contain rich harmonic content. It shouldn't be a pure sine, for example, as in that case there would technically be only one frequency present and the chance that there would be a similar frequency in the modulator is quite slim, and in that case the output would be more or less silent.

Besides standard vocoding as described, MVocoder also contains additional modes, which mingle the spectra of the carrier and modulator in different ways. An example would be morphing, which takes the 2 inputs and depending on the Ratio parameter keeps more of the one or the other.




## Presets

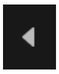
Presets button shows a window with all available presets. A preset can be loaded from the preset window by double-clicking on it, selecting via the buttons or by using your keyboard. You can also manage the directory structure, store new presets, replace existing ones etc. Presets are global, so a preset 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

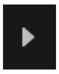
Files are named based on the name of the plugin like this: "{pluginname}.presets", so for example MAutopan.presets or MDynamics.presets. If the directory cannot be found on your computer for some reason, you can just search for the particular file.

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.

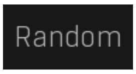


### Right arrow

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.

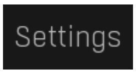
In addition, there are some mouse modifiers that assist this process. The smart randomization engine is used by default if no modifier keys are held.

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.

### Panic

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




### Settings

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

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

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

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

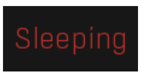
**Global system settings** contains some settings for all MeldaProduction plugins. Once you change any of them, restart your DAW if needed, and it will affect all MeldaProduction plugins.

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

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

### WWW

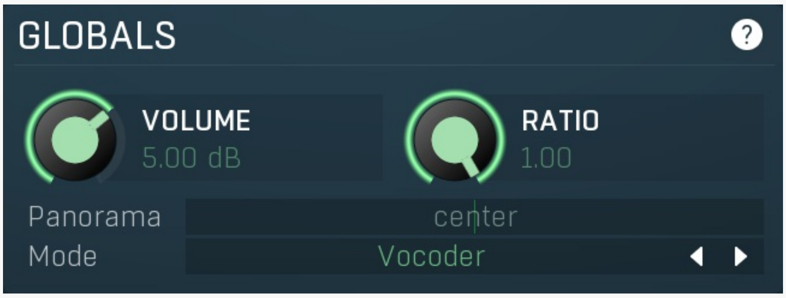
WWW button shows a menu with additional information about the plugin. You can check for updates, get easy access to support, MeldaProduction web page, video tutorials, Facebook/Twitter/YouTube channels and more.



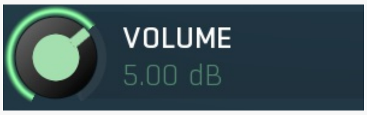

### Sleep indicator

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.

# Globals panel

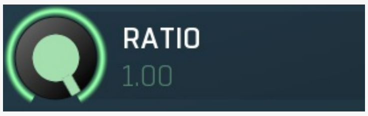



Globals panel contains global parameters such as vocoder mode, volume and panorama.




**Volume**

Volume defines the output volume of the processed signal.
Range: silence to 12.0 dB, default 0.00 dB




**Ratio**

Ratio is an additional parameter for the operation performed on each band and its meaning depends on the **Mode** parameter.
Range: 0.00 to 1.00, default 1.00


**Panorama**

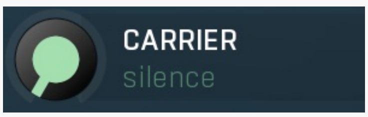
Panorama defines the output panorama of the processed signal.
Range: 100% left to 100% right, default center


**Mode**

Mode controls the actual operation performed on each band.
**Vocoder** mode performs the typical vocoder processing - the carrier signal is multiplied by the envelope of the modulator. Ratio controls the amount of processed signal.
**Dual vocoder** mode is similar to Vocoder mode, however Ratio controls amount of reverse-processed signal - modulator multiplied by the envelope of the carrier.
**Morph 1 and Morph 2** modes morph between the carrier and modulator signals in some way. The basic idea is to take the dominant signals from both of them. Ratio controls which one, carrier or modulator, is preferred.
**Ring modulation** mode performs per-band ring modulation, which can often result in highly inharmonic sound and can be used to create special effects. Ratio controls the amount of the processed signal.
**Exciting** mode amplifies the bands dominant in both carrier and modulator signals. Ratio controls the amount of processed signal.
**Inversion and Dual inversion** modes are similar to Vocoder and Dual vocoder modes, but the envelopes are inverted, so instead of keeping the frequencies in carrier which are dominant in modulator they keep the frequencies which are not dominant in modulator.

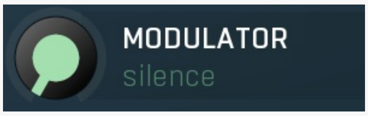
# Carrier & modulator panel


Carrier & modulator panel contains parameters controlling routing and direct carrier & modulator output.




**Carrier volume**

Carrier volume defines the output volume of the carrier signal (the main input by default).
Range: silence to 0.00 dB, default silence




**Modulator volume**

Modulator volume defines the output volume of the modulator signal (the side-chain input by default).
Range: silence to 0.00 dB, default silence


**Carrier pan**

Carrier pan defines the output panorama of the carrier signal (the main input by default).
Range: 100% left to 100% right, default center


**Modulator pan**

Modulator pan defines the output panorama of the modulator signal (the side-chain input by default).
Range: 100% left to 100% right, default center


**Swap carrier and modulator**

Swap carrier and modulator exchanges the carrier and modulator signals. By default the carrier (usually synth) is the main input and modulator (usually voice) is the side-chain.


**LR encoding**

LR encoding exchanges the standard side-chain input for input stereo - the left channel becomes the carrier, the right channel the modulator.

# Effects panel


Effects panel contains parameters of some effects performed by the vocoder.


**Whitening**

Whitening controls the amount of special effect applied to the carrier signal, which basically amplifies missing frequencies according to the detector results, hence resulting in a richer sound.
Range: 0.00% to 100.0%, default 0.00%


**Formant shift**

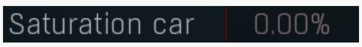
Formant shift lets you manually alter the formant information (in semitones), which generally results in no pitch shifting, but can create the mickey-mouse effect for example.
Range: -12.00 to +12.00, default 0


**Gate threshold**

Gate threshold controls the per-band noise gate threshold.
Range: silence to 12.0 dB, default silence


**Gate ratio**

Gate ratio controls the per-band noise gate ratio. The higher it is, the steeper and rougher the gate is.
Range: 1.0 : 1 to 20.0 : 1, default 2.0 : 1



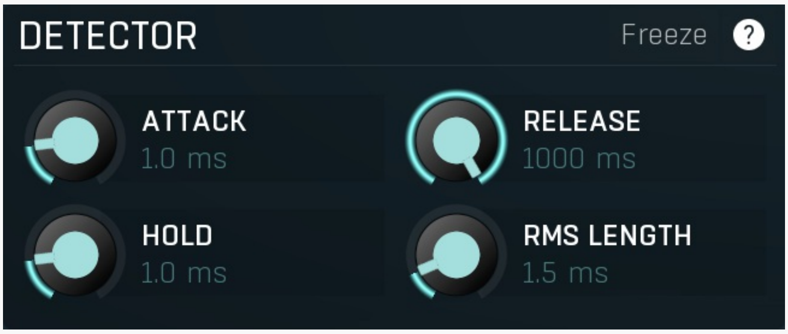

**Saturation car**

Saturation car controls the amount of saturation performed on the carrier signal. Saturation adds higher harmonics making the spectrum richer, which could be advantageous for vocoding.
Range: 0.00% to 100.0%, default 0.00%


**Saturation mod**

Saturation mod controls the amount of saturation performed on the modulator signal. Saturation adds higher harmonics making the spectrum richer, which could be advantageous for vocoding.
Range: 0.00% to 100.0%, default 0.00%

# Detector panel




Detector panel contains envelope detector parameters.


**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.*
Range: 0 ms to 1000 ms, default 1.00 ms


**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.

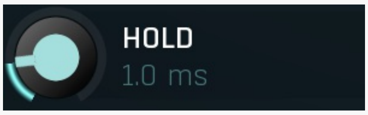
To understand the working of a level detector, it is best to cover the typical cases:

*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.*
Range: 0 ms to 1000 ms, default 10 ms

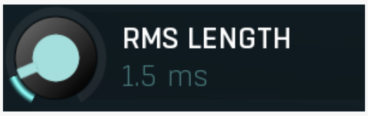



### Peak hold

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.
Range: 0 ms to 1000 ms, default 1.00 ms




### RMS length

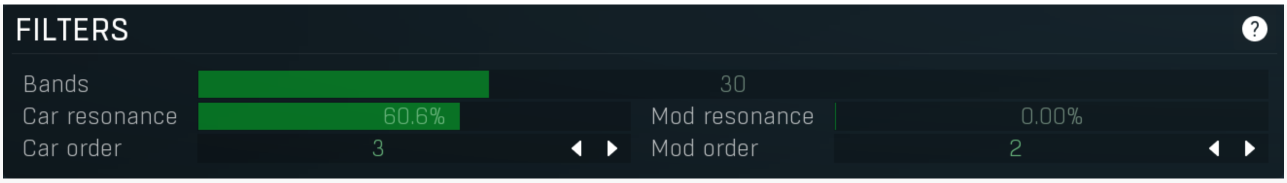
RMS length smoothes out the values of the input levels (not the input itself), such that the level detector receives the pre-processed signal without so many fluctuations. When set to its minimum value the detector becomes a so-called "peak detector", otherwise it is an "RMS detector".

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.
Range: Peak to 100 ms, default 1.0 ms

## Filters panel




Filters panel contains advanced parameters regarding the filters being used by the vocoder.


**Bands**

Bands controls the number of bands used by the vocoder. More bands usually provide higher clarity and higher CPU usage.
Range: 4 to 100, default 32


**Car resonance**

Car resonance controls the filter resonance applied to the carrier signal. A low resonance results in a fuller sound, while a higher resonance ends in a thinner sound.
Range: 0.00% to 100.0%, default 0.00%


**Mod resonance**

Mod resonance controls the filter resonance applied to the modulator signal. A low resonance results in a fuller sound, while a higher resonance ends in a thinner sound.
Range: 0.00% to 100.0%, default 0.00%


**Car order**

Car order controls the filter order applied to the carrier signal. Higher order provides higher clarity and higher CPU usage.
Range: 1 to 10, default 3

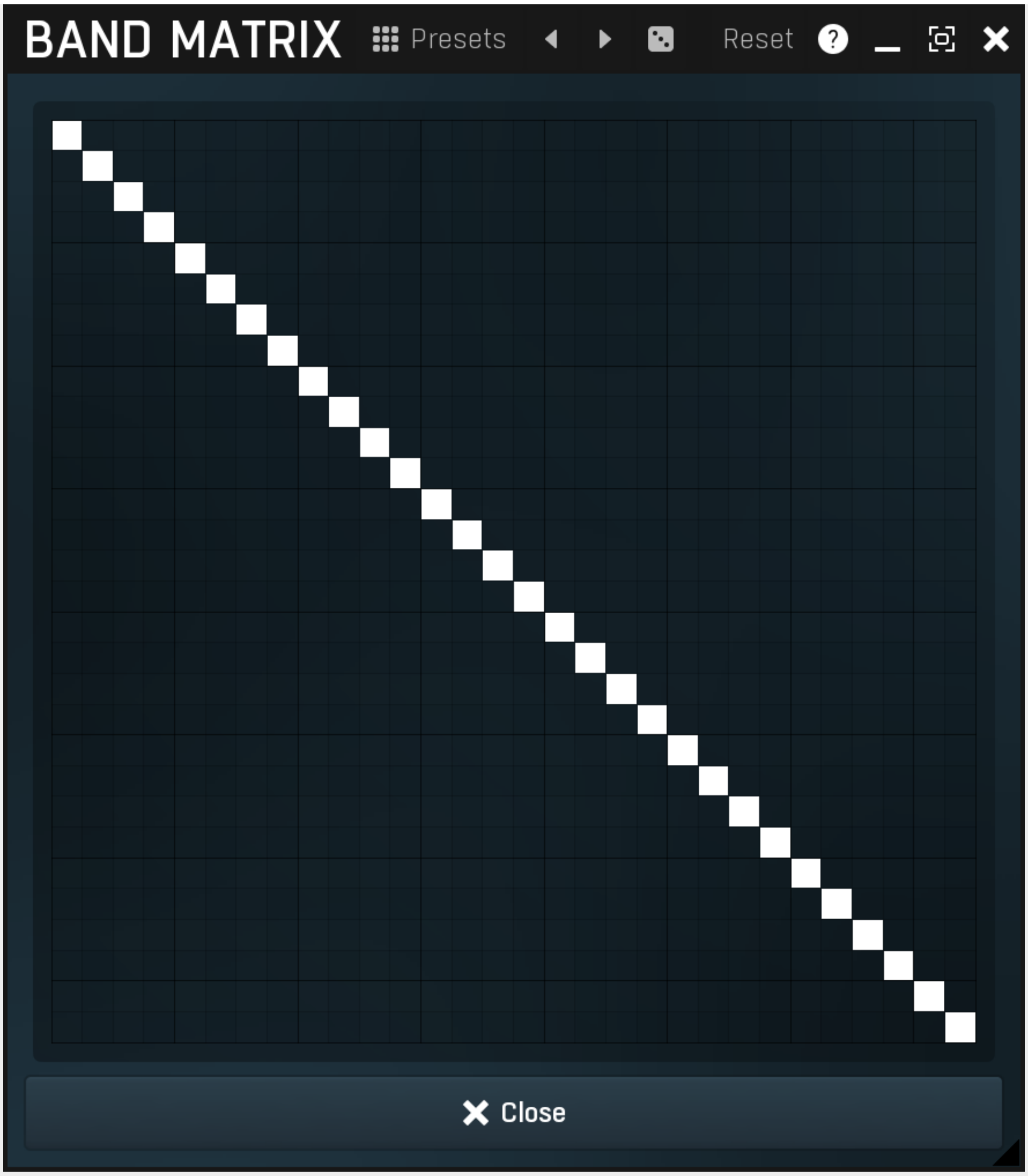

**Mod order**

Mod order controls the filter order applied to the modulator signal. Higher order provides higher clarity and higher CPU usage.
Range: 1 to 10, default 2

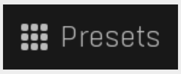

**Matrix**

Matrix button displays a band matrix, which controls how the bands affect each other.

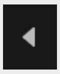
# Band matrix



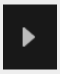

Band matrix controls how the bands affect each other. By default the graph is a diagonal line from top left to bottom right, which means that modulator band 1 affects carrier band 1, modulator band 2 affects carrier band 2 etc. However you may set this up differently and you may even let some carrier bands be affected by multiple modulator bands, or none of them.



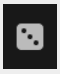

**Presets**
Presets button displays a window where you can load and manage available presets. Hold **Ctrl** when clicking to load a random preset instead.



**Left arrow**
Left arrow button loads the previous preset.



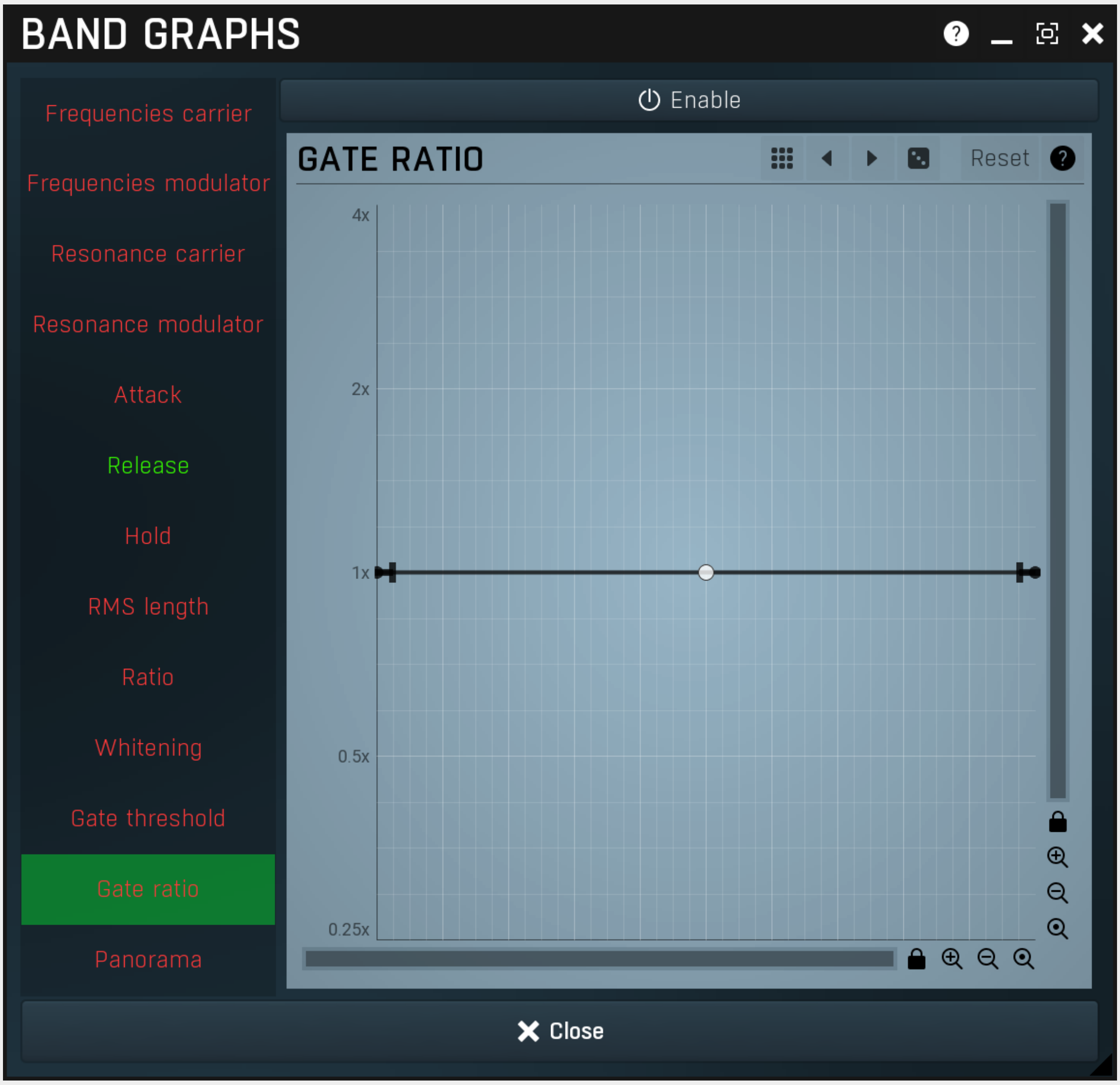
**Right arrow**
Right arrow button loads the next preset.



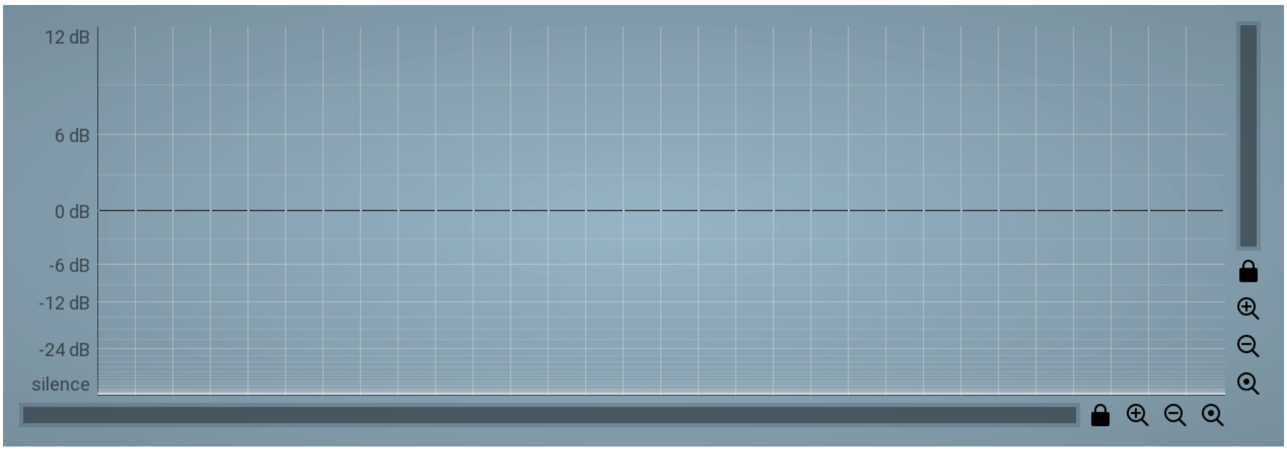
**Randomize**
Randomize button loads a random preset.


**Graphs**
Graphs button displays a set of graphs, which you can use to control features of each band, such as the distribution of the bands along the frequency axis, resonances, dynamic properties etc.

# Band graphs window




Band graphs window provides a set of graphs, which you can use to control features of each band, such as the distribution of the bands along the frequency axis, resonances, dynamic properties etc.

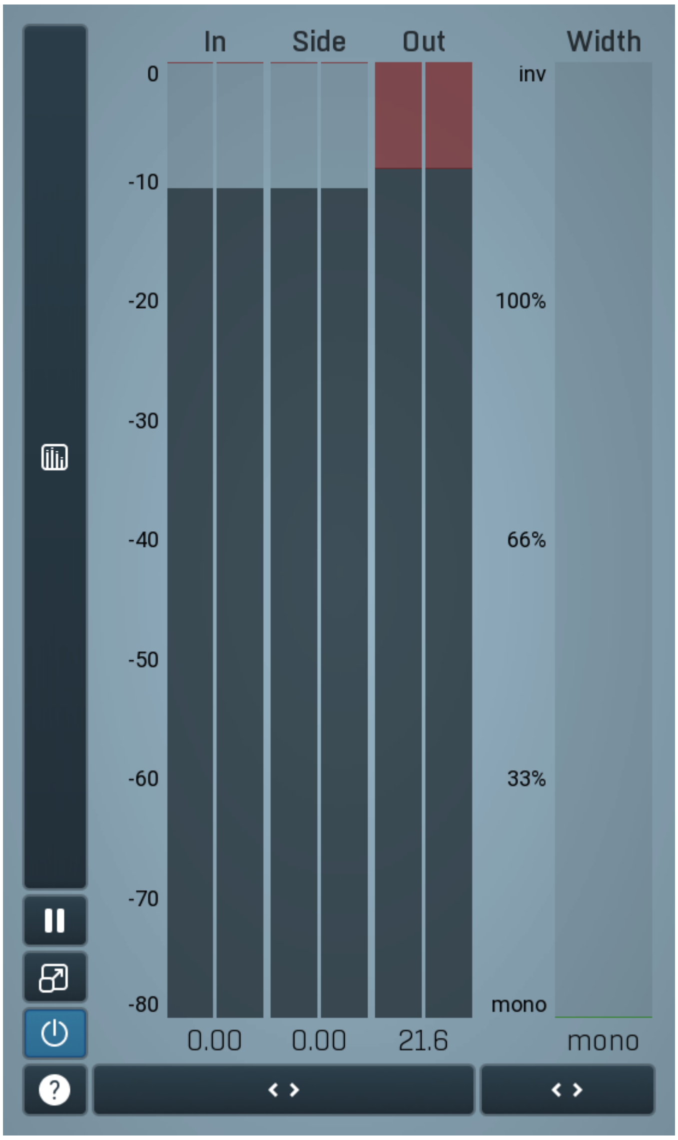



## Equalizer

Equalizer contains volumes for each band.

## Collapse

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




## 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.

**In meter** indicates the total input level. The input meter shows the audio level before any specific processing (except potential oversampling and other pre-processing). It is always recommended to keep the input level under 0dB. You may need to adjust the previous processing plugins, track levels or gain stages to ensure that it is achieved.

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

**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.

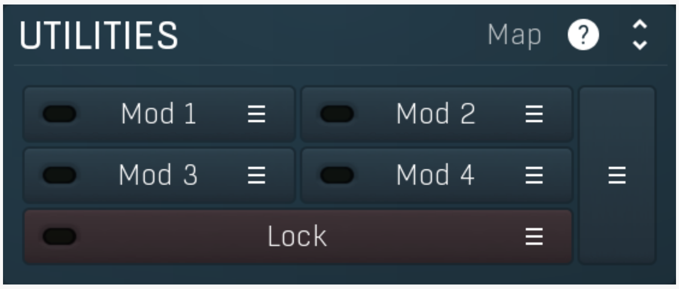
### Collapse

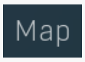
Collapse button minimizes or enlarges the panel to release space for other editors.

**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).


**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.

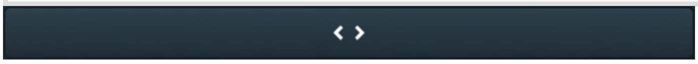
*For example, to associate a frequency slider and make a modulator control it from 100Hz to 1KHz, just enable the smart learn mode, click the slider then move it from 100Hz to 1KHz (you can also edit the range later in the modulator window too). Then disable the learning mode by clicking on the button.*

**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.


**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

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