d16 group

SIGMund

High Quality Flexible Delay Unit

User Manual

Overview

Sigmund is a multi-tap delay type effect with four identical and independent delay lines, which can be connected together in a few different manners (work in a few different configurations).

After loading the plug-in to the host application the following graphic interface appears.

Sigmund's graphic interface

The following elements can be distinguished:

- **Delay parameters** – parameters section of the chosen delay line (one of four).

Delay parameters section

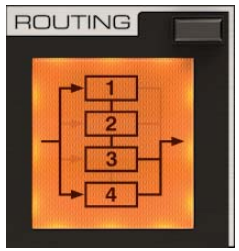
- **MOD1 / MOD2** – two **Modulators** of general use, which can influence certain internal parameters of delay lines.

MOD1 and MOD2 section

- **Mixer** – part of the interface which controls the process of mixing the delay lines' signal outputs.

Delay lines mixer

- **Routing** – control, which enables choosing one of **nine** available configurations/connection topologies of delay lines.



Control enabling the choice of the connection topology of delay lines

- **Master** section – allows to apply the final touch to the processed sound by the plug-in before passing the sound on to the audio output.

Master Section

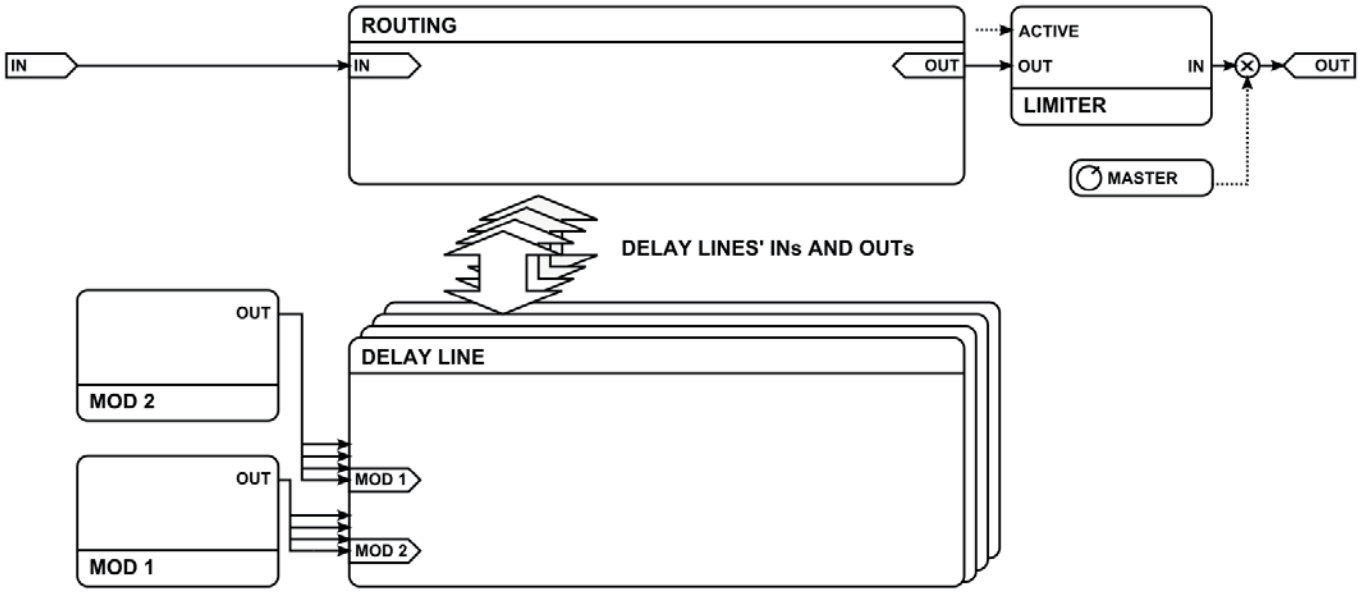
- **Preset** section – allows to choose/load presets, save current settings as a new preset or overwrite them on the currently chosen preset. The preset section also provides access to the **Preset browser**, which allows to perform more advanced operations on the preset files.

Preset management section

Signal flow

This chapter focuses on the plug-in signal's processing route, presents and describes its components and their controlling parameters situated in specified sections of the graphic interface.

The general diagram of the signal's processing by the plug-in is pictured below:

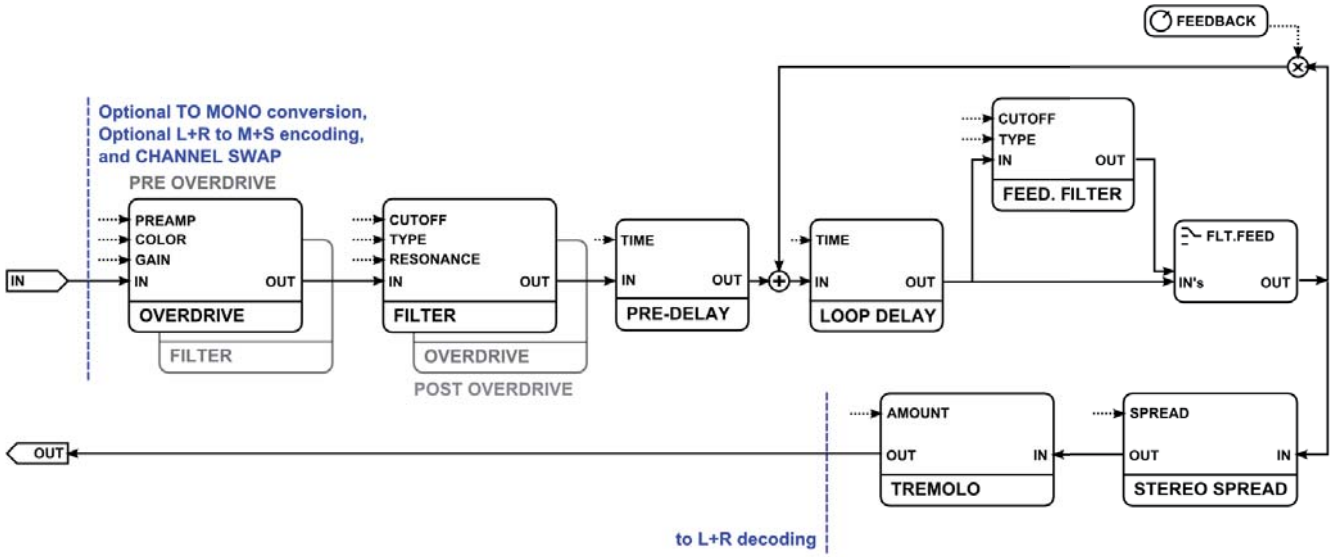


Plug-in signal flow general block diagram

The signal is transferred from the input to the router module, from which it is sent TO particular delay lines and BACK, enabling the forming of different connection topologies between those lines. For example, the signal can be processed parallel through delay lines, serial (passing on from the output of one line to the input of the next line), or in a mixed manner, in which some delay lines are connected parallel and others serial. After the signal reaches the end of the route set by router, it is transferred to the plug-in output after passing through the limiter.

Delay line

The most important element of the entire effect is the delay line module. As it was mentioned before, there are four delay lines available and every line has the same set of parameters.



Signal flow block diagram along a single delay line

The above picture presents a circuit diagram of Sigmund's delay line, in which the signal, while passing through the router, passes in sequence through the following modules:

- **Overdrive** distortion,
- **Multimode resonant filter** (or inversely, the sequence is variable),
- Initial signal delay (**Pre-Delay**),

- Subsequently, the signal enters the feedback loop and is processed successively by:
  - Feedback delay module – **Loop Delay**,
  - Optionally by the **Passive Feedback Filter**.
- A certain part of the signal (defined by the **Feedback** parameter) returns to the beginning of the loop (which causes an echo effect or a resonance in case of very small loop delays).
- By leaving the loop, the signal enters the module which moves one of the **Left / Right** channels (or **Mid / Side** depending on the given delay line's channel mode) in phases of provided time interval – **Stereo spread**.
- Subsequently, the signal goes through the amplitude modulator – **Tremolo**.

The parameters responsible for controlling the delay line are situated in the **Delay Parameters** section on GUI:

Delay parameters section

Choosing a delay line for editing is possible by using **Sel.** / **Copy** buttons in the **Mixer** section.

Mixer section – Sel. / Copy buttons

The buttons work in radio mode and the backlights inside the buttons and numbers in the upper left corner of the **Delay parameters** section indicate which line is currently chosen.

Delay parameters section – Delay line number

Multimode resonant filter and distortion module

Firstly, the signal which is passed on the delay line goes through the **Multimode resonant filter** and the **Overdrive** distortion module:

Multimode resonant filter and Overdrive subsections

Multimode resonant filter

Multimode resonant filter is situated in the **Filter** subsection:

Resonant filter parameters

The **Multimode resonant filter** is controlled by the following parameters:

- **Type** – Type of filter:
  - **OFF** – Inactive filter,
  - **LP** – Low-pass filter,
  - **BP** – Band-pass filter,
  - **HP** – High-pass filter.
- **Cutoff** – Filter cutoff frequency (or band frequency for **BP**).
- **Reso** – Resonant filter lining (or band width for **BP**).

Overdrive distortion module

Overdrive distortion paramaters

Overdrive distortion module is controlled by the following parameters:

- **Preamp** – signal amplification before distortion using the symmetrical diode-clipper.
- **Color** – controls the distortion type; a band with additional harmonics appearing, as a result of signal distorting. A low value of **Color** parameter results in a subdued sound and additional harmonics appear in the low and medium band. The higher the value of the parameter, the more harmonics appear in the higher bands and less in the lower and medium bands.
- **Gain** – controls the output volume of the distortion module.

The button in the upper left corner of the subsection activates/deactivates the **Overdrive** module.

Both modules described above (**Resonant filter** and **Overdrive**) are situated in the delay line just before the feedback loop's summing node. Therefore, before the signal is processed in the delay line, it is processed by both aforementioned modules.

Using the **Pre** and **Post** buttons (working in radio mode) in the **Filter** subsection:

Pre and Post resonant filter work mode

Pre and **Post** resonant filter work mode determines the sequence in which the input signal is processed by both modules:

- **Pre** (Pre Overdrive) – The signal is initially passed on the **Resonant filter** and subsequently on the **Overdrive** distortion module.
- **Post** (Post Overdrive) – The signal is initially distorted (**Overdrive**) and subsequently filtered.

Feedback loop's passive filter

In the next part of the sequence there is the passive filter in the feedback loop, which type and cutoff frequency are controlled by the same parameters as the **Multimode resonant filter** (parameters of both filters are connected). By default, the filter in the loop is inactive and it can be activated by the **Feedback** button (working in toggle mode) in the subsection of the **Filter** section.

Activating feedback filter in feedback loop

In this case the signal is additionally filtered by the **Passive filter**, apart from being filtered by the **Resonant filter** before passing on to the feedback loop. Apart from the influence of the **Feedback** parameter from the **Delay** subsection, as a result of which the signal amplitude is getting lower, this results in the fact that during each loop, the signal is more subdued (when **LP** is set), or with a more limited band (when **BP** is set) or it has less low frequency harmonics (when **HP** is set). If the chosen **Filter type** is **OFF**, then both the **Multimode resonant filter** and the **Feedback filter** in the feedback loop are inactive.

Pre-delay and Feedback delay

The successive processing elements in the delay line are:

- **Pre-delay** module – moving the signal by a provided time interval before it will be passed on to the feedback loop.
- **Feedback delay** module – controls the signal delay in the feedback loop, before it returns to the loop's beginning and will be summed with the input signal.

Parameters of both components are controlled in the **Time** subsection of the **Delay Parameters** section.

Time subsection

The **Delay** and **Pre** buttons are used to indicate whether the value of **Pre-delay** or the value of **Feedback delay** will be edited in the **Time** subsection.

Choice between editing Pre-Delay or Feedback loop delay

Delay synchronized with tempo

The delay time can be synchronized with the host application tempo. The synchronization is activated with the **Tempo sync** button:

Activating delay with tempo synchronization mode

Changing its state changes also the content of the **Time** subsection i.e. the set of available controls/parameters. The synchronization with the tempo can be activated independently to the **Pre-delay** or **Feedback delay**.

When **Tempo sync** is active, the value of the **Time** section looks as follows:

Time subsection view in synchronization with the tempo mode

The display of the section shows the current values of delays for both channels (**L** and **R** in **Stereo** mode or **M** and **S** in **Mid** / **Side** mode) in the form of common fractions. On the left side the delay of the **Left** (**Mid**) channel is displayed, on the right side the delay of the **Right** (**Side**) channel is displayed:

Display with delay values – synchronization with tempo mode

Dragging the mouse arrow over the numerators and denominators allows to change their value independently, influencing the rhythmical values of channel delays. **Full**, **Dot** and **Triplet** buttons on both sides of the display are modifiers of rhythmical values set on the display:

Rhythmic values modifiers – synchronization with tempo mode

- **Full** – Full note, in accordance with the display.
- **Dot** – Note with a dot.
- **Triplet**

Delays not synchronized with tempo

When the **Tempo sync.** is inactive (no synchronization with the host) the **Time** subsection content looks as follows:

Time subsection view in not synchronized with tempo mode

In this mode the delays are set in milliseconds in the range from 0.1 to 1000. The values of delays for the **Left** and **Right** channel (**Mid** / **Side**) can be changed independently by using the potentiometers on both sides of the display:

Value of delays for both channels – not synchronized with tempo mode

Alternatively, the desirable value can be precisely set by modifying a specific item after dragging the mouse arrow over the delay values represented on the display.

Tempo tap

The tempo tap allows to quickly set the delay time by repeatedly pressing the **Tap** button (for **Left** and **Right** channel). The time measured between subsequent clicks is averaged and afterwards set as the delay line time for a respective channel

Tempo tap button

Join channels mode

Using the **Join channels** mode allows to set the delay times for both channels (**Left** and **Right**) simultaneously. A change of the **Left** channel delay will immediately cause an identical update in the **Right** channel delay value and vice versa. This mode is activated by the **Join channels** button in the **Time** subsection and works both for the **Pre-delay** and **Feedback delay** regardless of whether it is active in synchronization with tempo mode or not (**Tempo sync.** flag).

Join channels mode

Delay line feedback parameters

Delay line feedback parameters are edited in the **Delay** subsection:

Delay subsection

Three parameters are available:

- **Feedback** – which can be set as a positive or negative feedback with zero value in the center:
  - **Zero value** – lack of feedback, the sound is passed on once by a provided time interval of the loop delay module and does not return to the beginning of the loop.
  - **Positive feedback** – The signal delayed by the loop delay module is added to the signal on the loop input proportionally to the value of that parameter. The larger the value of the **Feedback** parameter, the larger part of the signal is passed on the line input causing longer sound fading (more echoes).
  - **Negative feedback** – works similarly to the positive feedback, however the delayed signal is subtracted from the signal on the feedback loop input proportionally to the value of the **Feedback** parameter. As it was in the case of positive feedback, the higher the value of the parameter, the longer period of time the sound fades (as a result of more echoes).
- **Spread** – a phase shift between **L+R** channels (**Left** and **Right** channel) or **M+S** channels (**Mid** and **Side** channel). This parameter may have the following values:
  - **C** (**Center**) – The center value is neutral, meaning no phase shift.
  - **L** (**Left**) – Along with decreasing the value of the parameter towards **L**, the phase shift between channels (Stereo; **L+R**) is increasing, meaning that channel **L** precedes channel **R**. Minimal value i.e. maximal inclination towards **L** means that the phase transition between channels takes half the time of the feedback loop delay.
  - **R** (**Right**) – Along with decreasing the value of the parameter towards **R**, the phase shift between channels (Stereo) is increasing, meaning that channel **R** precedes channel **L**. Minimal value i.e. maximal inclination towards **R** means that the phase transition between channels takes half the time of the feedback loop delay.
- **Hold** – button working in monostable mode, which, when pressed, causes a momentary setting of the **Feedback** value to **100%**. If the **Feedback** value is positive, then it is set to **+100%**. If the **Feedback** value is negative, then it is set to **-100%**.

In **M+S** (**Mid** / **Side**) mode, **S** (**Side**) replaces the maximal value of **R**, and **M** (**Mid**) replaces the minimal value. The rule is the same as in the case of channels **L+R**, but the **Spread** is a phase delay between **Mid** and **Side**.

Processing in Left/Right, Mid/Side and Mono mode

Every delay line in Sigmund processes the stereo signal in **L+R** (**Left** and **Right** channel) mode, therefore by setting delays (**Pre-delay**, **Feedback delay**) or by moving channels toward each other (**Stereo spread**), the operations are performed in the **L+R** domain. Sigmund also allows to process the signal by a delay line in the **M+S** (**Mid** and **Side** channels) domain. In this case, delays (**Pre-Delay**, **Feedback delay**) are set for **M** and **S** channels instead of **L** and **R** and the same applies to the phase shift of channels in the feedback loop (**Stereo spread**), the shift is made between **M** and **S** channels. Whether the processing should be performed in **L+R** or **M+S** mode can be set per delay line by choosing the adequate mode using **Channel mode** buttons:

- **L / R** – the line works in **Left+Right** mode
- **M / S** – the line works in **Mid+Side** mode

Additionally, by using the **Ch. Swap** button channels can be swapped with each other (**L** with **R** or **M** with **S**, depending on the value of **Channel mode**).

Choice between L+R and M+S processing

Swapping channels (L with R or M with S)

To mono button is used to monophonize the delay line's input signal.

Monophonizing the input signal

The button works in toggle mode; pressing the button activates the option, pressing it again deactivates the option. The operation consists of summing the input signal from the left and right channels and overwriting them with a resultant signal.

Operations described in this subsection are performed at the beginning of the delay line in the following order:

- Monophonizing (optional)
- Choosing the processing mode (**L+R** or **M+S**)
- Swapping channels (**L** with **R** or **M** with **S**, depending on the chosen mode)

Therefore, when **To mono** mode is activated and subsequently, the channel swap option is used (**Ch. Swap**) while the **M / S** mode is on, the signal received on the output will not contain the **M** component, only the **S** component, regardless of the signal provided on the input (stereo or mono).

Delay line parameters modulation

In Sigmund there are two **Modulators** of general use available. However, these are not **Modulators** independent for every delay line, but global generators with the parameters defined for the entire instance. The **Modulators** can be used to modulate some of the delay line parameters.

MOD1 and MOD2 sections

The range of influence and the parameters of a given delay line which are influenced by the **Modulator** are defined per delay line and these values can be different for every delay line.

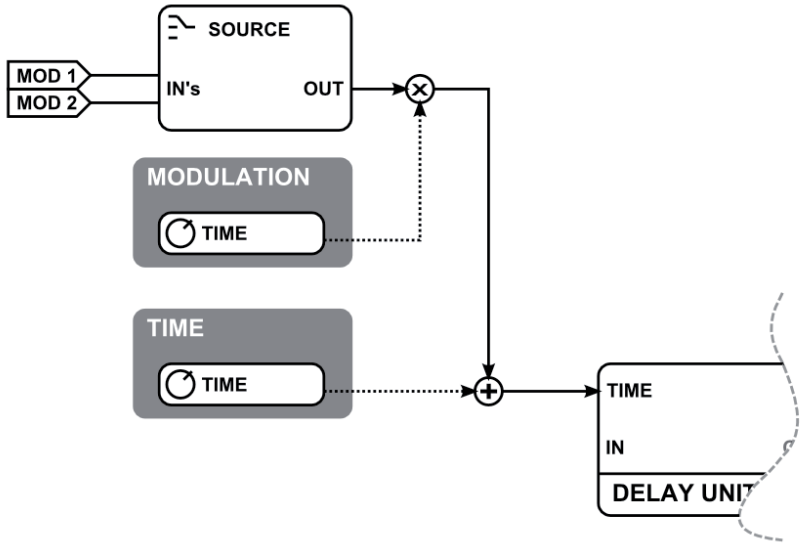
The **Modulation** subsection in the **Delay Parameters** section allows to define the influence of the chosen **Modulator** on three parameters of the delay line:

- **Delay time in the feedback loop**
- **Cutoff frequency** of delay line filters
- Delay line's output signal **amplitude**

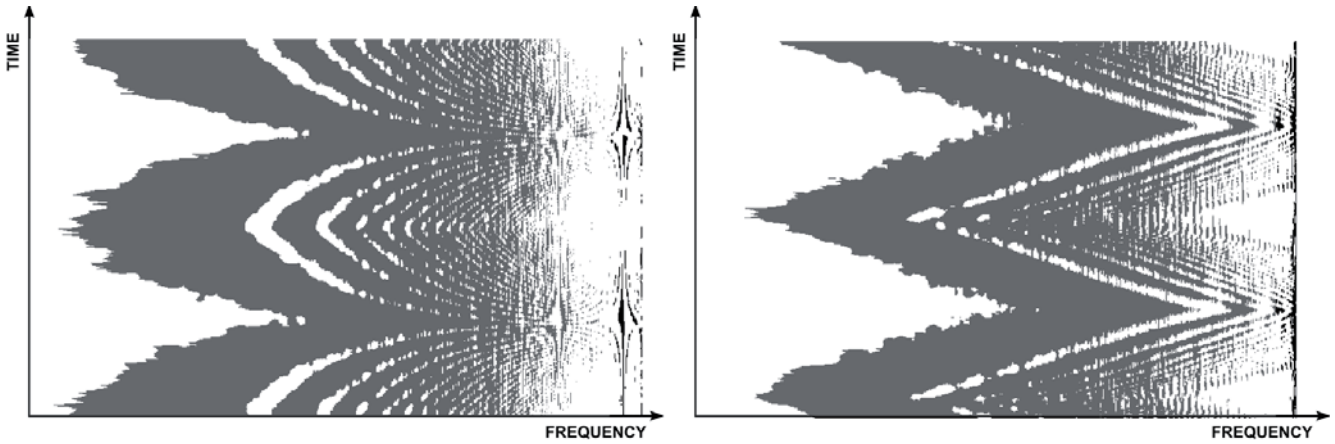
In the **Modulation** subsection the following parameters are available:

Modulation subsection

- **Time** – range of influence of the **Modulator** on the delay value in the feedback loop. **Lin** and **Log** buttons, next to the **Time** parameter, work in radio mode and they allow to choose a scale according to which the **Modulator** should influence the delay:
  - **Lin** – Linear scale
  - **Log** – Logarithmic scale

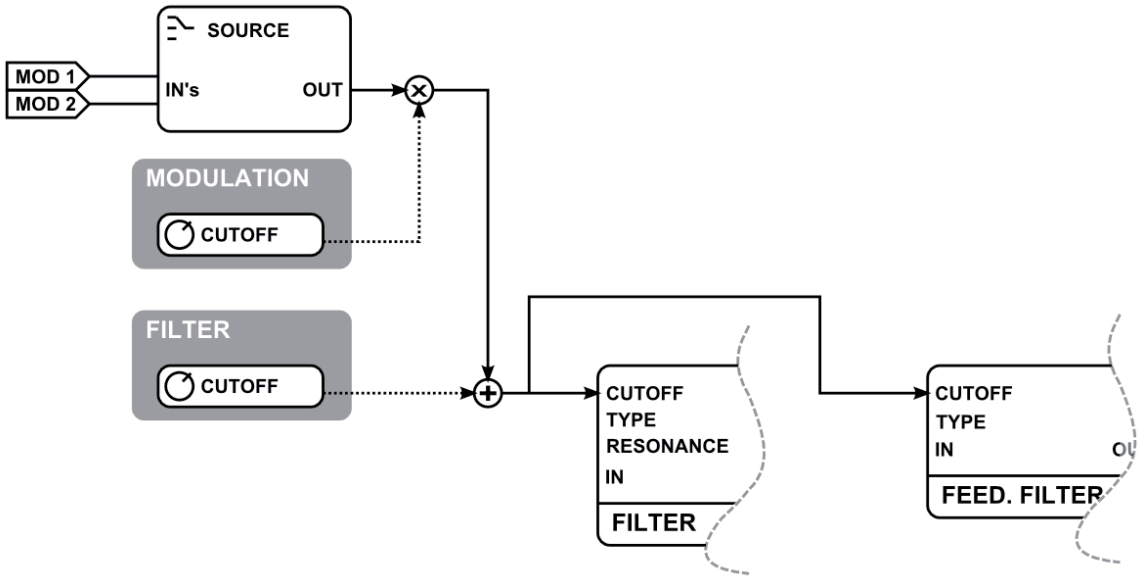


Block diagram of delay modulation



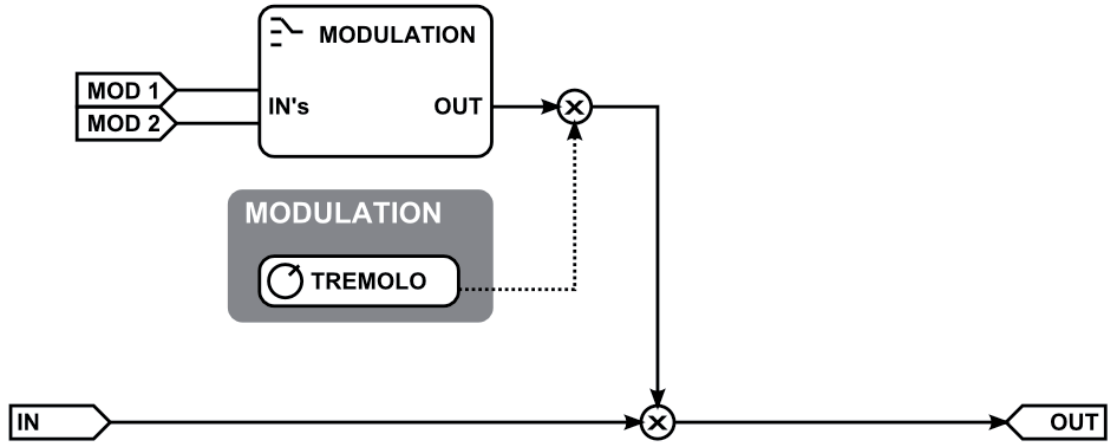
Example of influence on the value of loop delay in linear scale on the left, and in logarithmic scale on the right in a case of Modulator working as LFO (Spectogram is scaled logarithmically)

- **Cutoff** – range of influence of the **Modulator** on the cutoff frequency of the **Resonant filter** and the **Passive filter** in the feedback loop (the resultant value, being the sum of the **Modulator's** output and the value of **Cutoff** parameter, is provided in the **Filter** section as the input cutoff frequency, which is the same for both filters in the delay line (**Feedback** and **Resonant**).

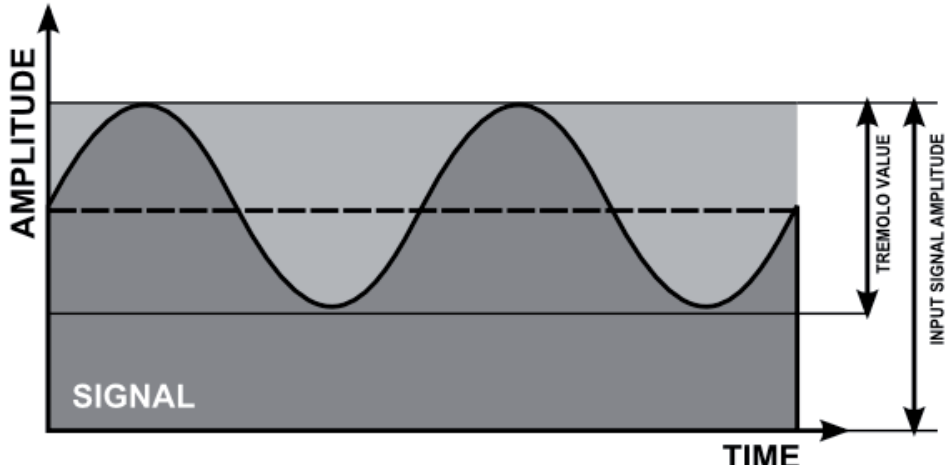


Block diagram of cutoff filter frequency modulation

- **Tremolo** – range of influence of the **Modulator** on the output delay line's amplitude. The higher the value of the parameter, the more the minimum of the generated **Modulator** route nears to the minimum off the modulated signal.



Block diagram of amplitude modulation – Tremolo Effect



Influence of Tremolo parameter on the depth of amplitude modulation by LFO type Modulator

MOD1 and **MOD2** buttons, working in radio mode, allow to chose the source of modulation (one of the two available low frequency oscillators):

Choice of the source of modulation for delay line

Modulators

The two available in Sigmund **Modulators** of general use can be used to modulate some of the delay lines' parameters (**Amplitude**, **Filter cutoff time**, **Delay time** in feedback loop).

MOD1 and MOD2 sections on GUI

Each of the **Modulators** can be of three types:

- **LFO** – Low frequency oscillator
- **ENV** – Envelope
- **PEAK** – Peak follower

The **Modulator** type can be selected by the radio group on the left:

Modulator type selection

LFO Modulator

LFO can work in synchronization with the host application tempo or without it, dependently on the **Sync** button status. The default mode is without synchronization. Switching the **Sync** parameter slightly changes the view of the **LFO** section on the GUI because only part of the **LFO's** parameters is common for both modes.

LFO without synchronization with the host application tempo

LFO without synchronization with the tempo

Without the **LFO's** synchronization with the tempo, the following parameters in the **Modulator** section on GUI are available:

- **Frequency** – Frequency of the generated oscillations:

Frequency of LFO oscillations

Additionally, the **Frequency** diode flashes accordingly to the **LFO** oscillation's frequency, helping to determine the rate of the subsequent cycles.

- **Wave** – Shape of the generated oscillations' wave (starting from the top):

LFO's waveform

  - **Triangle**
  - **Sine**
  - **Square**
  - **Falling sawtooth wave**
  - **Smoothed Sample and Hold (S/H)**
- **Invert** – Reversing the phase of the generated oscillations

Wave inverting

- The **LFO** generates waveforms independently for both stereo channels. **Phase** parameter controls the phase shift between channels (**Left** and **Right** or **Mid** and **Side**, dependently on the selected delay line signal representation). The range of this parameter is from **0** to **180** degrees, in which zero means lack of phase shift (both channels have an identical waveform), whereas **180** degrees means generating the stereo signal in a counterphase

LFO Phase shift

LFO in synchronization mode with the host application tempo

The **LFO** synchronization with the host application tempo is activated with the **Sync** button.

Activating synchronization mode with tempo for LFO

If the **Sync** mode is active, the oscillation frequency **Rate** is defined in rhythmic values dependent on the tempo (in the range from **32** to **1/32**time). Otherwise, the oscillation frequency is defined in Hz (from the range of **0.01** Hz to **40** Hz).

When the **Sync** mode is active, the **LFO** section's view slightly changes:

View of LFO section with synchronization with tempo mode active

Additional buttons appear (working in radio mode), which allow to choose one of three available rhythmical **Rate** value modifiers:

Modifiers of LFO rhythmical scale in synchronization with tempo mode

- **Full** – The rhythmical value remains unaltered (in accordance with the value of the **Rate** slider).
- **Dot**ted – The rhythmical value is increased by 1.5 times the value of **Rate**.
- **Tri**plet – The rhythmical value constitutes 1/3 of the value of **Rate**.

Resetting the LFO phase

Regardless of whether the **Sync** mode is active or not,the **LFO** phase can be optionally reset; externally with a MIDI note or internally with transients detected in the input signal (for detailed information refer to **Modulator (re) triggering** chapter).

The moment of resetting the **LFO** phase is indicated by the Reset diode in the upper-left corner of the section.

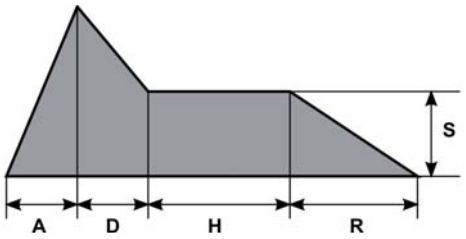
Reset Diode in the LFO section

Envelope Modulator

Envelope Modulator

The **Envelope** generator is another available **Modulator** type in Sigmund controlled with the following set of parameters:

- **A**ttack time
- **D**ecay time
- **S**ustain level
- **H**old time
- **R**elease time



Envelope modulator in Sigmund

Envelope triggering

In order to activate the **Envelope** it must betriggered with a MIDI note or transients detected in the input signal (for detailed information refer to **Modulator (re)triggering** chapter).

The **Trig.** diode, in the upper-left corner of the **Envelope** section, indicates the moment of triggering with a MIDI note or a detected transient.

Envelope section's Trig. Diode

Peak Follower Modulator

Peak Follower Modulator

The last type of the **Modulator** is the **Peak follower**, which allows to trace input signal's amplitude envelope by following the momentary maxima of the signal. The delineated envelope can be used to modulate the delay line's parameter.

The **Peak Follower's** is controlled by the following parameters:

- **Attack** – the amplidute's raising slopes following speed
- **Release** – the amplidute's falling slopes following speed
- **Sensitivity** – the amplification of the input signal for detection

Modulator (Re)triggering

Each of the two available **Modulators** of general use can be (re)triggered by one of the events:

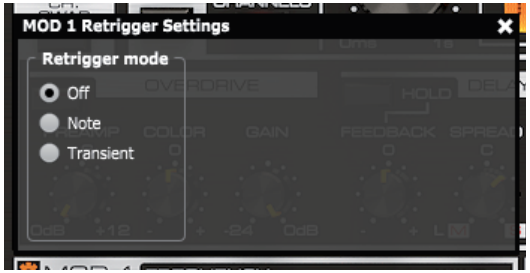
- **The occurrence of a MIDI note** with a specific pitch on a respective MIDI channel
- **Transient** detected in the input signal.

(Re)triggering is available for all types of **Modulators** except the **Peak Follower**, therefore:

- For the **Envelope**, which is triggered when an event occurs
- For the **LFO**, which phase is reset when an event occurs.

The type and parameters of the (re)triggering event are configured in the **Modulator** itself (independently in both). Clicking the sprocket icon in the upper-left corner of the **Modulator** section reveals the **Modulator Parameters**:

The icon which opens the Modulator Parameters section



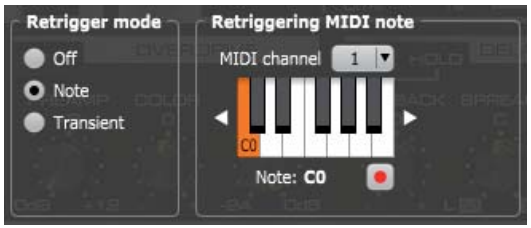
Modulator Parameters window

Retrigger mode defines the event type:

- **Off** – (re)triggering is inactive
- **Note** – the triggering event is the MIDI note
- **Transient** – the triggering event is the transient in the input signal

Depending on the selected event type, it's parameter values / conditions of occurrence are set on the right side:

- For **Note** mode – the **MIDI channel** must be defined (using the combo box) as well as the note's pitch (using the control – the graphic representation of the music keyboard), which will trigger the event:

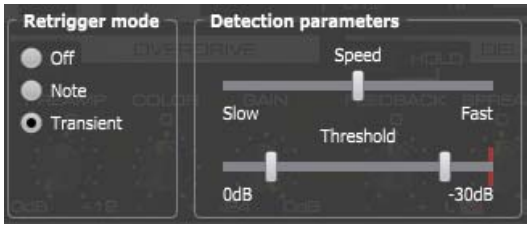


Re(triggering) with the midi note

The **Arm** button activates the **Learn** mode, in which the note's pitch can be quickly set using the connected and active MIDI keyboard.

Learn Mode

- For **Transient** mode:



Transient detection parameters

➲ *Note: The (re) triggering with the MIDI note works only in the VST version of the plug-in.In the AU version this functionality is not available.*

– **Speed** – the rate at which the envelope of the input signal is followed.

– **Threshold** – the moment when the transient appears; passing the signal's envelope through the **upper threshold** (left limiter), only if it has earlier passed through the **lower threshold** (right limiter). Transient is an occurrence of a rapid raising slope in the input signal's envelope. The red vertical line along the **Threshold** parameter slider represents the temporary value of the envelope, showing a visual feedback, which allows to comfortably select the **threshold** values.

Delay lines mixer

On the right side of the **Delay parameters** section there is the **Mixer** section, which is used to easily control the output volume of particular delay lines, set the panning for each of them or choose a particular line for edition.

Mixer section

The first row consists of **Sel.** / **Copy** buttons (working in radio mode), which are used for choosing a delay line for edition (in **Delay parameters** section).

Choice of delay line for edition

The next row consists of **Mute** / **Paste** buttons, which are used for immediate muting of delay lines. The buttons work in toggle mode, pressing the button (with the left mouse button) will mute the line, pressing the button again will restore the signal on the output of the particular line. The current state of the button is indicated by its backlight.

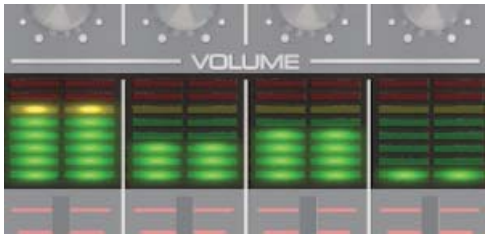
Muting of particular delay lines

Clicking with the right mouse button, **Mute** buttons work in **Solo** mode, will mute all the delay lines except the one, to which the pressed **Mute** button corresponds.

The next row consists of knobs controlling the **Panning** value of particular lines.

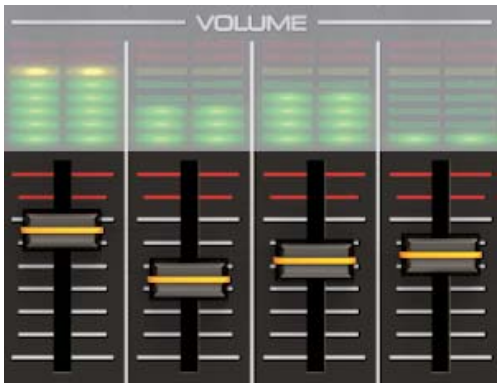
Panning setting for particular delay lines

The next row consists of VU-meters, which indicate the signal level on the outputs of particular lines.



VU-Meters measuring output signal level on delay lines

The last row in the **Mixer** section consists of faders controlling the signal volume on the delay lines' outputs.



Faders controlling output signal level on particular lines

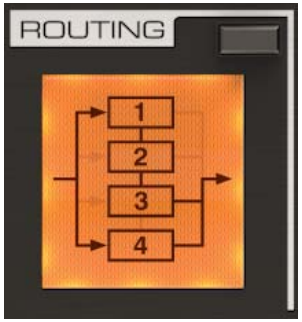
Copying delay lines settings

Using **Sel.** / **Copy** and **Mute** / **Paste** buttons enables copying the settings of one delay line onto the other:

1. Copy the delay (source) line parameters value to the buffer. Clicking **Sel.** / **Copy** button while holding **CTRL** button (on Mac OS X use **Apple CMD**) saves the parameters of the line corresponding to the pressed **Sel.** / **Copy** button in the buffer.
2. Overwrite the chosen delay line with the buffer content using the corresponding **Mute** / **Paste** button by clicking it while holding **CTRL** button (**Apple CMD** on Mac OS X).

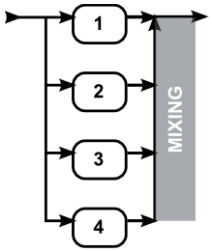
Routing

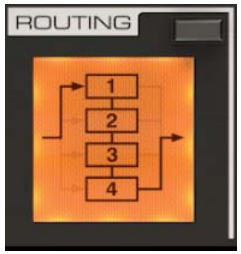
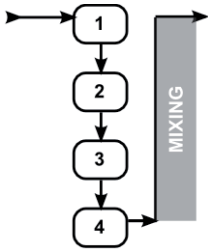
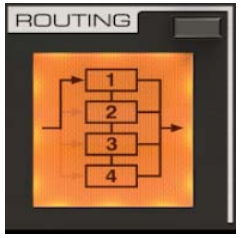
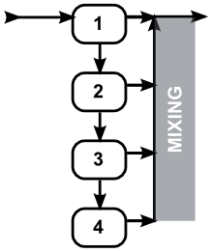
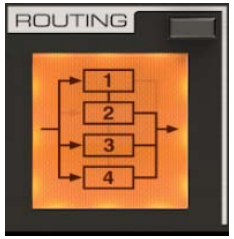
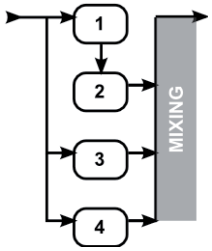
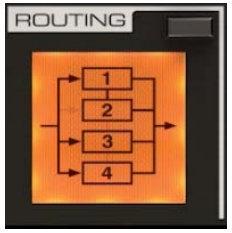
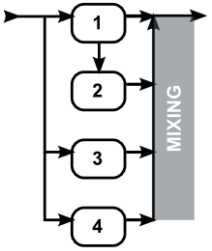
In Sigmund, delay lines can be connected in several different manners. The display in the **Routing** section shows the current configuration under which the delay lines work.

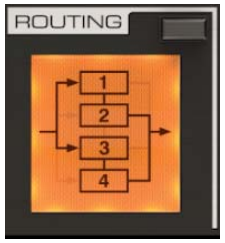
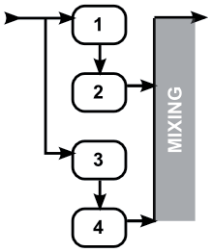
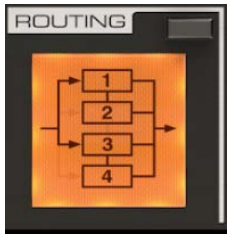
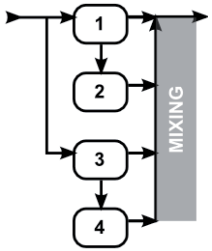
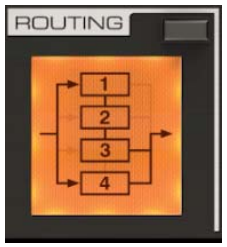
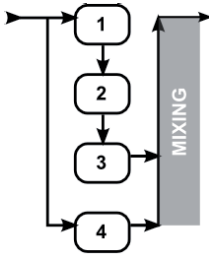
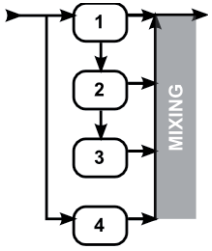


Routing section on GUI

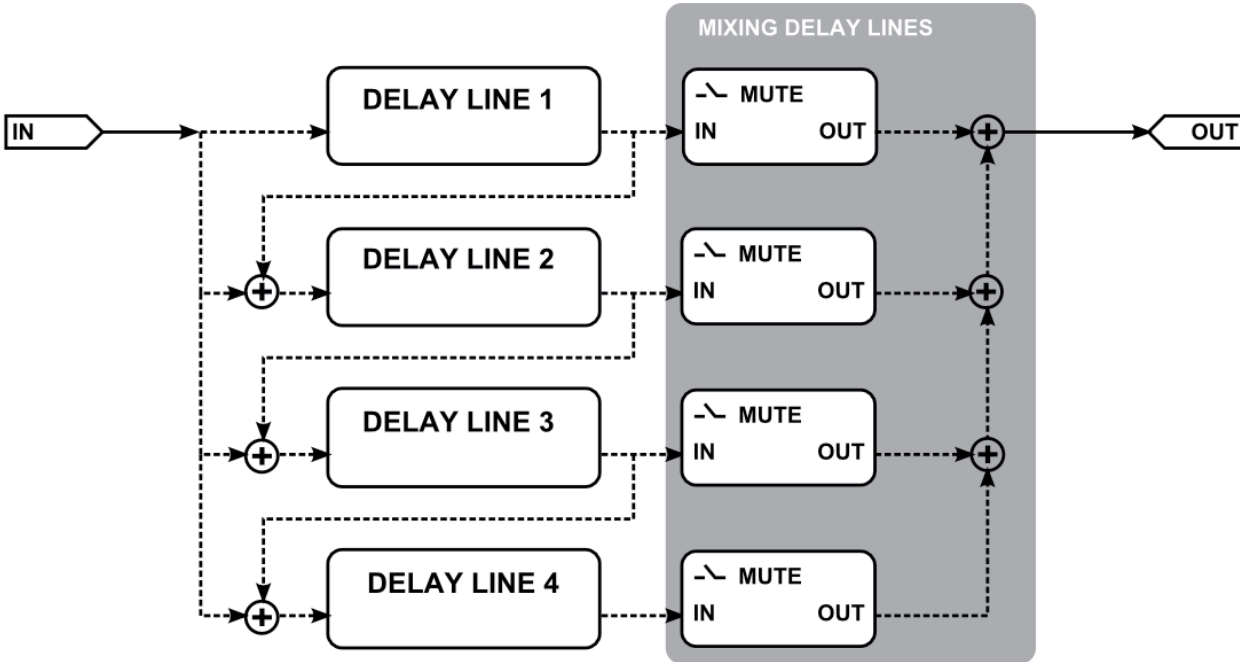
There is a button in the upper right corner of the **Routing** section, which opens a combo box with available connection configurations (**nine** different to choose from):

	GUI Symbol	Combo box name	Diagram	Description
1		Parallel		Parallel line connection (default). The signal is provided on the inputs of all four delay lines. Every line processes it independently and afterwards the line outputs are mixed together.

	GUI Symbol	Combo box name	Diagram	Description
2		Serial		Serial connection. The input signal is processed in a cascade manner by all delay lines and afterwards passed on the output.
3		Tapped serial		Tapped serial connection. The signal is processed in a cascade manner as in the previous configuration, but every line, besides passing its output on the input of the next line, also sends the signal to the mixer.
4		Mixed 1		Serial-parallel connection. First two lines work in a cascade manner, in which the output of the second line is send to the mixer, while lines 3 and 4 work in parallel mode and their outputs are passed on the mixer.
5		Mixed 2		Tapped serial – parallel connection. The configuration is almost the same as the previous one with one difference. Line 1 signal is tapped, therefore, apart from creating a cascade with line 2, line 1 passes the signal on to the mixer.

	GUI Symbol	Combo box name	Diagram	Description
6		Mixed 3		Semi-parallel connection (with two cascades). Lines 1 and 2 form a cascade, and lines 3 and 4 form a second one. Both cascades are processed parallel.
7		Mixed 4		Tapped semi-parallel connection (with two cascades). Similarly to the previous configuration there are two cascades, but the difference is that lines 1 and 3 are tapped and send their output to the mixer.
8		Mixed 5		Semi-parallel connection (with one cascade). Lines 1, 2 and 3 serially connected form a cascade, which works parallel to line 4.
9		Mixed 6		Tapped semi-parallel connection (with one cascade). Similarly to no. 8, but the difference is that lines 1 and 2 are tapped and send their output to the mixer.

It is necessary to add that in the case of a topology, in which the signal from one delay line is passed on the input of another line (cascade) and, additionally, the signal from line one is tapped and passed on to the **Mixer**, using the **Mute** button for line one mutes only the tap, which was directed to the **Mixer** input, while, regardless of the state of the **Mute** button, line one passes the signal on the second line.



Block diagram of Mute buttons functioning in Mixer section

The above picture presents all potential connections between delay lines in Sigmund. As it depicted, the taps are situated after the line output and before the **Mute** switch, therefore, using **Mute**, will not interrupt the signal flow from one line to the other when they are working in a cascade manner.

Master section

In the **Master** section the final touch is applied to the signal summed in the **Mixer** section.

Master section

The **Master** section contains the following elements:

- Pair of analogue-like **VU meters**, which observe the amplitude of the output stereo signal

VU-Meters in Master section

- **Limiter** with an automatic release

Limiter

in which:

– **Active** – switch responsible for turning on or turning off the limiter.

– **Threshold** – regulates the range of the output signal dynamics (top value, to which the input signal dynamics is “cut”).

- **FX** – parameter used to define the ratio in which the processed and unprocessed signal will be mixed by the effect before passing it on the output.

FX parameter (Dry/Wet)

- **Output** – parameter which controls the output signal level.

  Two **Left** and **Right** diodes, which inform about exceeding the maximal value of the amplitude by the output signal.

Parameter which controls output signal level

Preset Management

Preset storage

All kinds of presets are stored on a disk in a particular location, which makes the process of management smooth and allows to easily exchange presets between users. When you insert a plug-in to a host application, before the window opens, some initial actions are performed. Among those, Sigmund scans the location in which presets are stored on the hard disk and projects this location into the tree (hierarchical) structure, which corresponds to a structure of folders and files:

- Single files are projected as presets and the file name is the preset name
- A folder is projected as a group of presets, which is presented in the **Preset Browser** as a group (node). It is allowed to nest the structure of groups and single presets into another group etc.

In the **Preset** section we can always distinguish a few controls:

Preset management section

- The text box containing the name of the currently loaded preset.
- **Prev** / **Next** – These two buttons are used for linear navigation through the bank of presets.
- **Browse** – This button opens **Preset browser**.
- **Save** / **Save as** – Storing a current settings as a preset (overwriting/as new one).

For storing presets the standard OS dialogue windows are always used.

➲ *Note: All kinds of presets are stored accordingly to a human-readable XML standard. There is no need to edit presets manually, however, they can be opened and edited in any text / XML editor.*

By default, presets are kept on the hard drive:

- For MacOS X – the folder "~/Library/Application Support/D16 Group/Sigmund"
- For Windows – the folder "C:\Users\[user_name]\AppData\Roaming\D16 Group\Sigmund"

Sigmund recognizes .sgprst file extension as its preset.

Padlocks

Padlocks is a functionality which allows to block certain groups of parameters before changing their value during reloading presets. Blocking is possible before overwriting the parameters of particular delay lines (**Delay parameters** section) or/and value of the **FX** parameter (ratio between **Dry** and **Wet** signal). In order to block a group of parameters click the padlock icon. It works similarly to buttons in toggle mode, one click blocks a group of parameters, second click unblocks it.

Blocking delay lines parameters

Blocking delay lines parameters is achieved by using the padlocks situated in the upper part of the delay lines **Mixer**:

Padlocks which allow to block particular delay lines

In order to block parameters of a particular delay line, Simple click the padlock over the number corresponding to that particular line.

Every padlock encompasses the parameter values of a single delay line.

Blocking the FX parameter

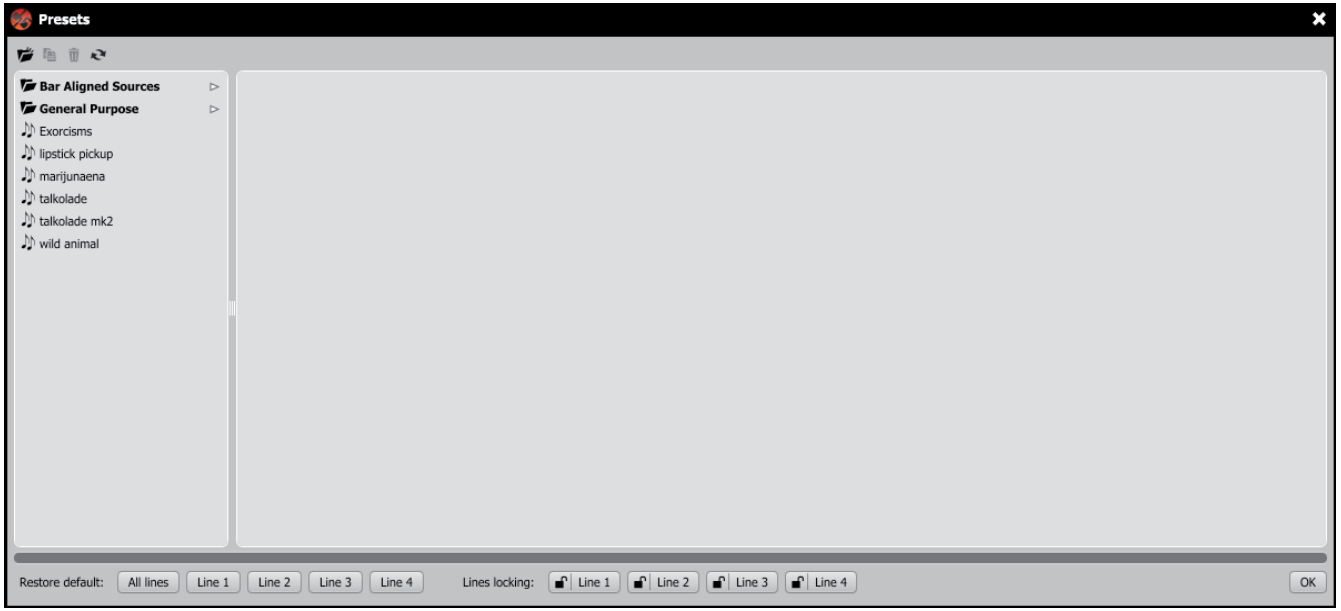
The padlock responsible for blocking the **FX** parameter is situated directly above it:

Padlock which blocks FX parameter

One click on the padlock is sufficient in order to block the parameter from overwriting its value while loading a preset.

Preset Browser

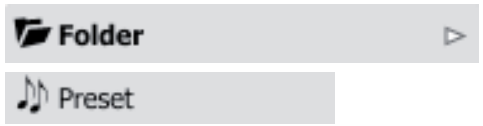
As it was mentioned in the previous chapter, presets in Sigmund are stored as files on the hard drive. Therefore, the **Preset Browser** reflects the file structure of a location on the disk where presets are actually kept.



Preset browser

The hierarchy of the file structure on the disk is represented by columns (each column is a single level in the directory tree), in which the leftmost column is the highest level in the file structure (**Preset** root folder), and subsequent columns, going to the right, represent successive levels of the directory tree. Each column may contain two kinds of items:

- **Folder** – Which name is emphasized with bold font and an arrow on the right.
- **Preset file**



Clicking the **Folder** selects it, and its content (subfolders and files) is displayed in the adjacent column to the right. If the folder we selected contains any subfolders, analogically you can perform the same action on and on, until reaching the deepest level of the directory tree. All columns starting from the left side will correspond to successive levels of the directory tree.

Clicking the **Preset** file selects and loads the preset, and also displays information about the preset in the column directly on the right side:

Preset Browser – Preset description

It contains information like: **Preset** name, **Author** and **Rating**. We can change the **Rating** according to our recognition by giving it **1** to **5** stars (by mouse clicking on one of the stars). The author's name is assigned automatically to all newly created and saved presets based on information given by user in a **Preset** tab in **Configuration** panel.

Double-clicking the item (**Folder** or **Preset file**) toggles edit mode, which allows to rename the file.

We can select (for removing or copying):

- A Single item by clicking it.
- Group of items:
  - By selecting the first one and adding new elements by selecting them while holding CTRL key (Apple CMD on Mac OS)
  - By selecting a range of items; click the first one (to mark the beginning) and then click the last one while holding CTRL key (to mark an end).

Selected item(s) can be moved to any subfolder using the drag'n'drop method. It is also possible to move a selected **Preset** / **Folder** or group level up in the directory tree using the drag'n'drop method, if we drop the selected item(s) onto the column on the right.

There are several buttons in the top bar of the **Preset Browser**. The buttons can be used to perform additional actions:

Preset Browser – Top Bar buttons

 – Creating a new **Folder** in the current location

 – Creating a duplicate of a selected **Preset** file (this option does not work on **Folders** and groups of items)

 – Deleting a selected item or group of items.

 – Refreshing the columns content in the **Preset browser**. It is necessary in the situation when Sigmund is loaded and the content of the hard drive (in the location where **Preset** files are kept) has changed.

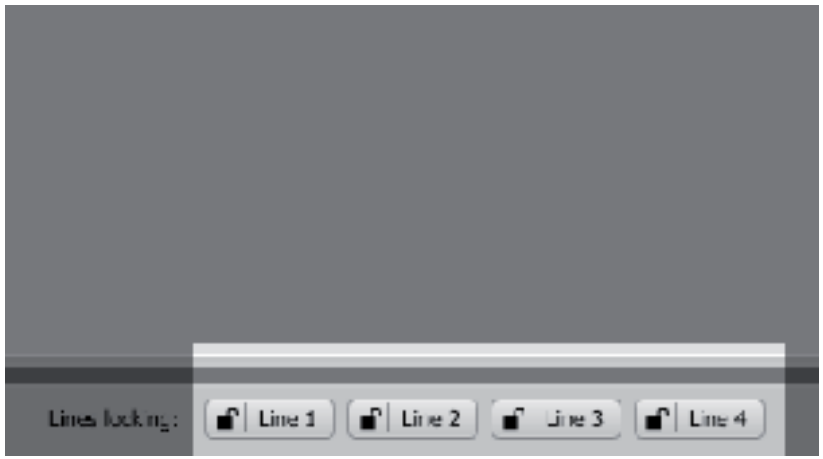
Preset Browser – Bottom Bar

The **Ok** button on the right or **Close** button in the upper right corner closes the **Preset Browser**.

Blocking delay lines parameters on the Preset Browser level

It is also possible to block delay lines parameters (use of padlock mechanism) before changing the value while loading a new preset without closing the **Preset Browser** window.

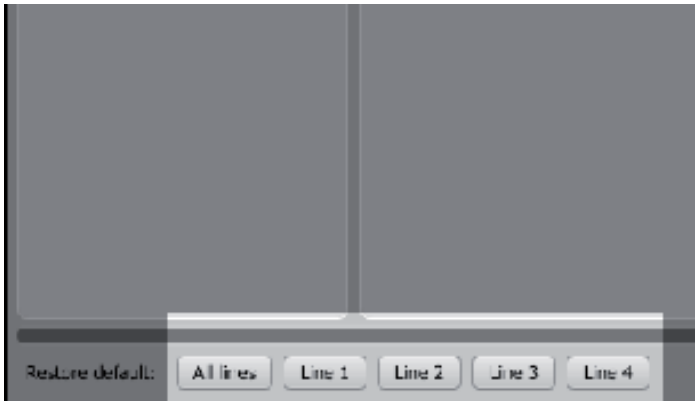
In the bottom part of **Preset Browser** there are padlock buttons, which work exactly in the same manner as those in the **Mixer** section i.e. in toggle mode. Clicking any padlock buttons blocks the delay line's parameters (with the number corresponding to the number on the buttons) before overwriting with new values while loading the preset.



Preset Browser – Padlocks, which block parameters of particular delay lines

Restoring default plug-in parameters values

In the bottom-right part of the **Preset Browser** window there are buttons situated, which enable restoring the default plug-in parameters values or parameters of any delay line.



Preset Browser – Restoring default parameters values

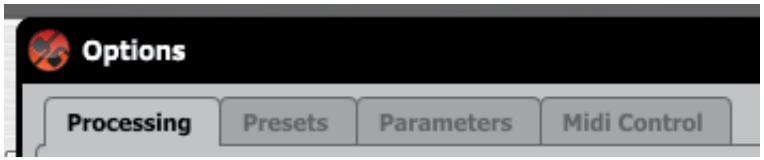
All lines button initializes all sound parameters with default values (current settings are lost), while buttons from **Line 1** to **Line 4** cause the initialization of delay lines **1** to **4** respectively. Default values mean the status of parameters just after setting the plug-in in the host application.

Configuration

The **Configuration** panel in Sigmund allows us to change the general settings of the plug-in. Use the **Options** button on GUI to load it:

Options / About button

In the **Configuration** panel the following tabs are available:

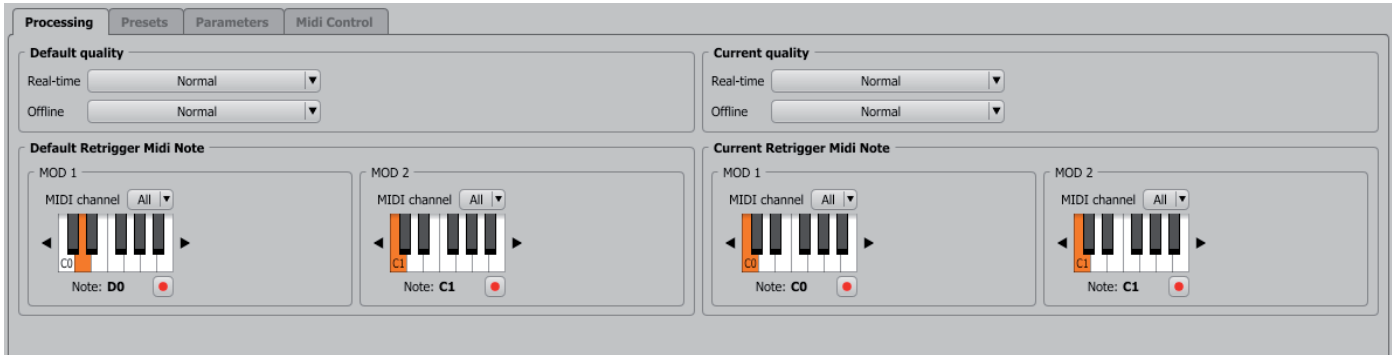


Configuration Panel – Tabs

- **Processing** – Global configuring the sound processing path.
- **Presets** – Global actions on preset files.
- **Parameters** – Configuring automatable parameters within the host application.
- **Midi Control** – Configuring MIDI communication with plug-in.

Processing

In the **Processing** tab, the current and default values of specific parameters influencing the processing path are configured.



Processing tab

Current quality settings

Current quality settings section

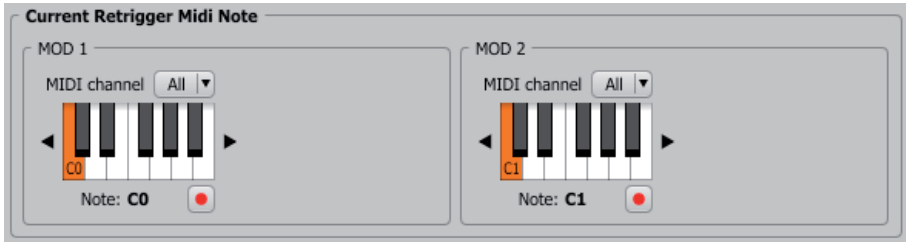
This section allows to set the processing quality of the plug-in while taking into consideration available system resources (CPU power). The grading includes four levels:

- **Low**
- **Normal**
- **High**
- **Ultra**

The qualities are arranged from the lowest to the highest and simultaneously from the least to the most demanding, considering the CPU load.

The processing quality selection is made independently for **Real-time** processing and **Offline** processing (e.g. during mixdown).

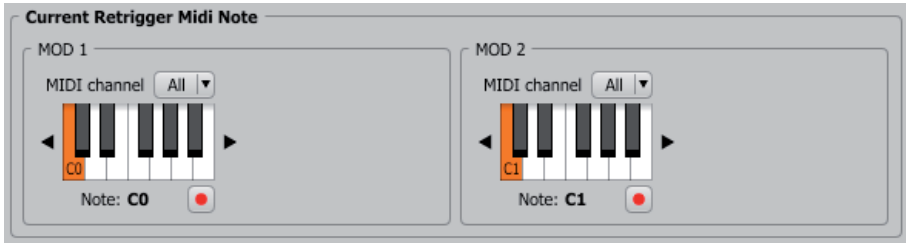
Current Retrigger MIDI Note settings



Current Retrigger MIDI Note section

The settings in this section are linked with the settings available in the **Modulator Properties** (see **Modulator (Re) triggering** chapter). They concern the (re)triggering the modulator with a MIDI note. The **Current Retrigger MIDI Note** section allows to configure (independently for each **Modulator**) the pitch and channel number of the MIDI note, which will retrigger the respective **Modulator** (provided that the **Retrigger Mode** parameter in the **Modulator Properties** is set to **Note**). **MOD1** and **MOD2** subsections allow to define the note's pitch and channel for **Modulator 1** and **Modulator 2** respectively.

Similarly as in the **Modulator Properties**, the **Learn** mode can be used to quickly define the MIDI note's pitch. Simply use the **Arm** button in the bottom-right corner of the contol:



Learn mode activation

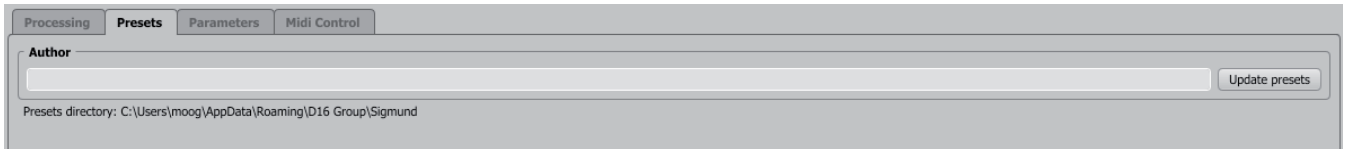
and set the note's pitch with one key press on the connected and active MIDI keyboard

Default settings

Every time a Sigmund is loaded in the host application (new instance is created) the **Default settings** value is used for a **Current settings**. **Default settings** are stored within a configuration file of Sigmund. This file is saved at the moment of unloading any of active plug-in instances from the host application.

Presets

In the **Presets** tab, we have an access to functions allowing certain global actions on preset file.



Presets tab

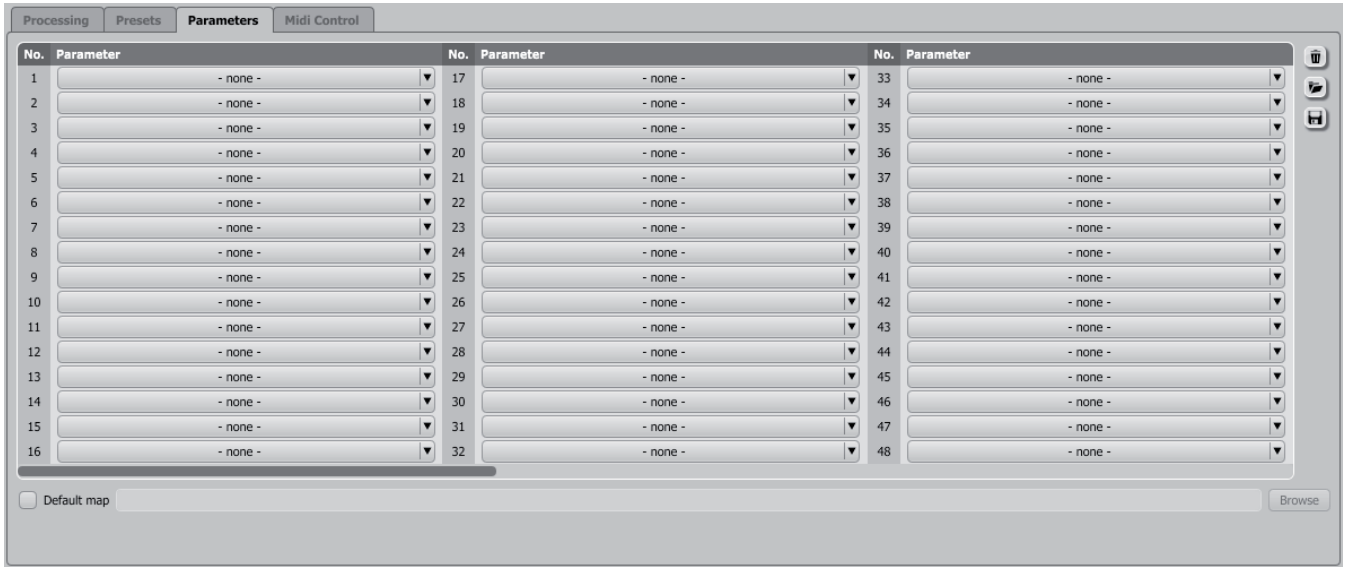
Every preset in Sigmund apart from storing the values of sound parameters, stores also certain additional information i.e. author's name or rating. The **Author** box in the **Presets** tab, describes the author's name which is going to be saved within every new instance of the preset. Double-click and edit the box, confirm with enter button to change the value of the box. Any new instance of the preset is going to encompass information from that box.

Update presets button updates information about the author in every new instance of the preset created by the user. Other presets, e.g. default presets, presets of other authors, will not be modified.

The **Presets directory** value below the **Author** box displays the path to the present bank on the hard drive.

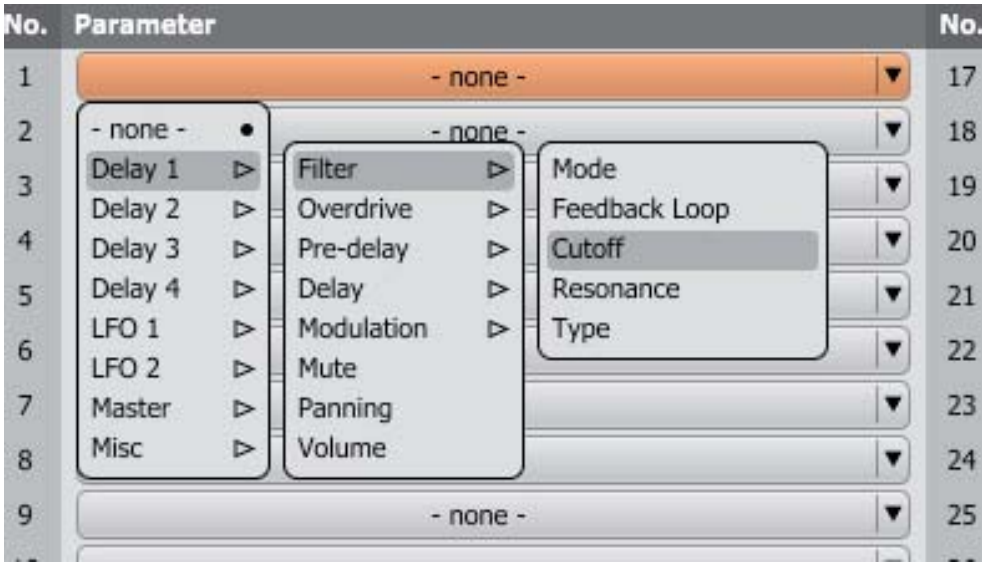
Parameters

Due to the fact that the number of sound parameters in Sigmund exceeds 128 allowed by VST / AU technologies number of automatable parameters, we can map/assign the chosen internal parameters of the synthesizer on 128 automatable (from the host level) general usage parameters in the **Parameters tab**.



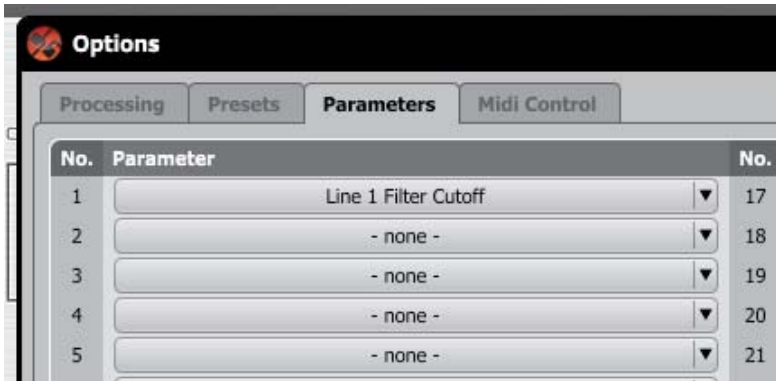
Options – Parameters tab

The parameter map is represented in the form of table, in which the column No. is the number of the general usage automatable parameter and the column **Parameter** is the internal parameter of the plug-in. Each line assigns an internal parameter to an automatable parameter. We assign an internal parameter using the combo box in the **Parameter** column.



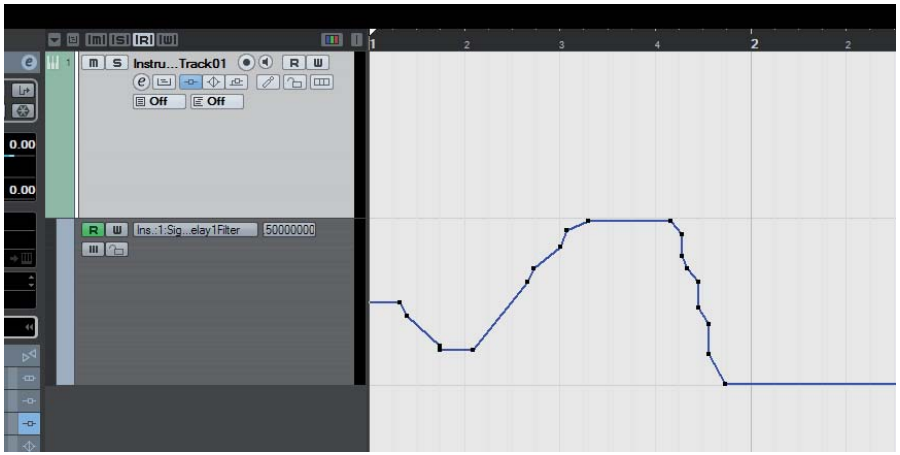
Host parameter assignment

For example, when we assign the general usage parameter No.1 to the internal delay line parameter **Delay 1** → **Filter** → **Cutoff**.



Parameters tab – Parameter mapping

Then we can automate the chosen parameter from the host application level.



Cubase – Delay 1 Filter Cutoff automation

Assignment List Management

➲ *Note: Assignment list files are compatible with XML format and can be viewed and edited in any text editor.*

On the right side of parameter assignment list there are editorial function buttons available:

 – Clearing the entire assignment list.

 – Loading the assignment list from a file (.sghpmap)

 – Saving the assignment list to a file (.sghpmap)

Default Parameters map

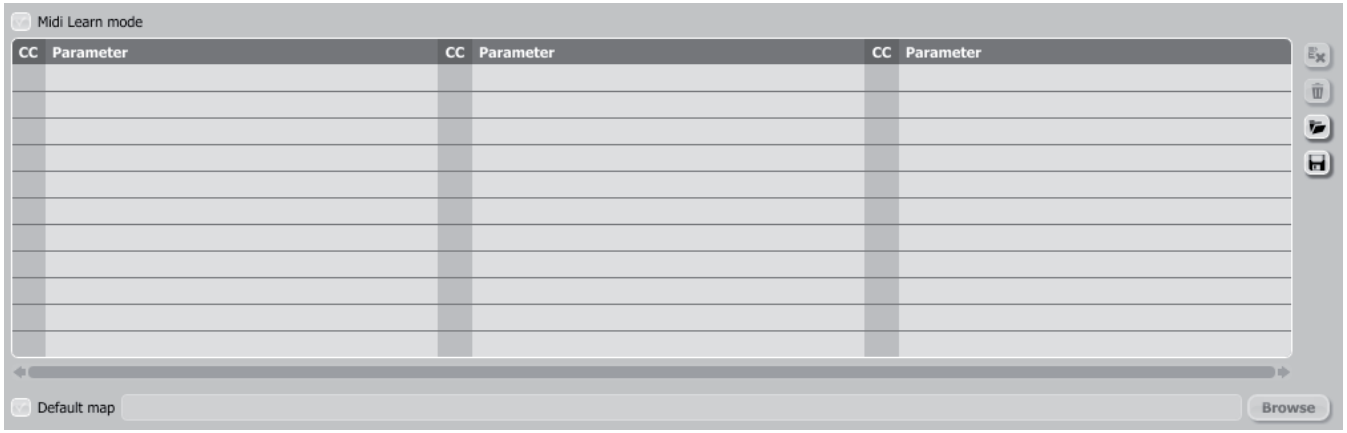
Selecting a default **Parameters** map:

- Check the **Default Map** checkbox, which activates the **Browse** button on the right.
- Click the **Browse** and select a file with saved **Parameters** map.

After selecting a **Parameters** map the text box on the left from the **Browse** button shows the path to the active map file. A default **Parameters** map is loaded each time when the plug-ins is loaded.

Midi Control tab

Sigmund can assign its controls (on GUI) to any MIDI CC (Midi Control Change) event, allowing the control of the plug-in to use external hardware or software controller.



MIDI CC Assignments table and functional button

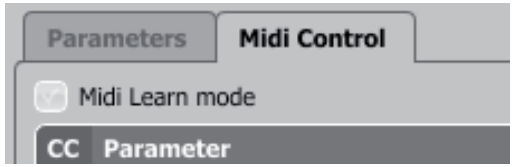
Controls included in the tab:

- **Midi learn mode** – Checkbox which activates **Midi learn** mode.
- **List of active MIDI CC assignments** – Containing pairs consisting of MIDI CC number and the name of the plug-in parameter.
- **Default Map** – Checkbox which activates a default MIDI CC map. When the map is activated it will be loaded with creating a new instance of the plug-in.

Midi learn

Assigning a Sigmund's control to the MIDI controller requires:

1. Checking the **Midi Learn** mode checkbox in the **Midi Control** tab:



Midi Control tab – Midi Learn mode checkbox

or pressing the **Midi Learn** button in the **Master** section on GUI:

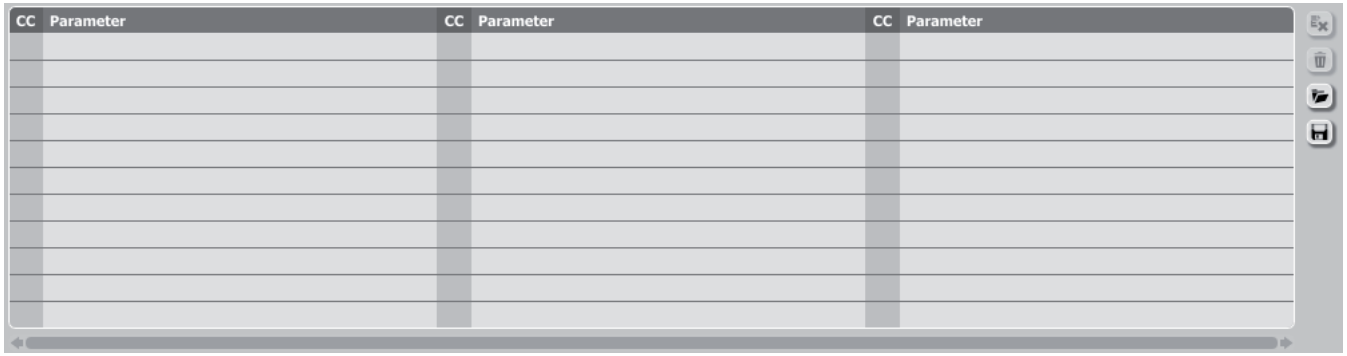
Master Section – Midi Learn

2. In the status bar in the bottom section of GUI a message should appear *"waiting for a controller or parameter movement..."*:

Status bar

In this mode the plug-in waits for any change in the parameter's value (movement of any control on GUI) and for a movement of any MIDI CC control on the external MIDI controller, which operates on an active MIDI input channel directed to Sigmund. The order of these actions is irrelevant.

During the above-mentioned actions, the status bar informs about the currently changing values of controls and provides their names.



MIDI CC Assignments table and functional button

When a link is established between MIDI CC and the sound parameter, a line is added to the list of MIDI CC assignments.

3. When all the needed links are created, uncheck the **Midi Learn Mode** checkbox or press again the **Midi Learn** button on GUI in the **Control** section.

In order to create new links, it is possible to reactivate the **Midi Learn** mode at any time.

The links are always sorted in an ascending manner in relation to the CC column (according to the MIDI CC code).

Unlinking and MIDI link management

On the right side of the link list there are four function buttons located:

Function buttons

 – Removes a selected link; selecting any link from the list and clicking on it highlights the selected link. Pressing this button removes the selected link.

 – Removes all MIDI CC links.

 – Loads a MIDI map from file (.sgccmap – Sigmund MIDI CC Map).

 – Saves all links in the list to a MIDI map file.

➲ *Note: MIDI map files are saved in XML format, which enables their edition in any text editor*

Default MIDI map

Selecting a default MIDI map:

Setting a default MIDI Map

1. Check the **Default Map** checkbox, which activates the **Browse** button on the right.
2. Click the **Browse** button and select a file with a saved MIDI map

After selecting a MIDI map the text box on the left from the Browse button shows the path to the active map file.
A default MIDI map is loaded each time when the plug-in is loaded.

Contents