



# **REAKTOR 5**

## **Operation Manual**

The information in this document is subject to change without notice and does not represent a commitment on the part of NATIVE INSTRUMENTS GmbH. The software described by this document is subject to a License Agreement and may not be copied to other media. No part of this publication may be copied, reproduced or otherwise transmitted or recorded, for any purpose, without prior written permission by NATIVE INSTRUMENTS GmbH, hereinafter referred to as NATIVE INSTRUMENTS. All product and company names are trademarks of their respective owners.

Furthermore, the fact that you are reading this text means you are the owner of legal version rather than an illegal, pirated copy. It is only through the loyalty and honesty of people like yourself that NATIVE INSTRUMENTS GmbH can continue to develop and create innovative audio software. On behalf of the entire company, thank you very much.

Users Guide written by: Rick Scott, Marius Wilhelmi, Len Sasso, Stephan Schmitt, Erik Wiegand, James Walker-Hall, Julian Ringel

Special thanks to the Beta Test Team, who were invaluable not just in tracking down bugs, but in making this a better product.

© NATIVE INSTRUMENTS GmbH, 2007. All rights reserved.



**NATIVE INSTRUMENTS**

**Germany**

NATIVE INSTRUMENTS GmbH

Schlesische Str. 28

D-10997 Berlin

Germany

[info@native-instruments.de](mailto:info@native-instruments.de)

[www.native-instruments.de](http://www.native-instruments.de)

**USA**

NATIVE INSTRUMENTS North America, Inc.

5631 Hollywood Boulevard

Los Angeles, CA 90028

USA

[sales@native-instruments.com](mailto:sales@native-instruments.com)

[www.native-instruments.com](http://www.native-instruments.com)

# Content Operation Manual

<b>1. Introduction .....</b>	<b>7</b>
1.1. What is REAKTOR? .....	7
1.2. New/Changed Features in REAKTOR 5.....	7
1.3. Event Initialization .....	7
1.4. REAKTOR Core Technology .....	8
1.5. New Primary Modules .....	9
1.6. Changed Primary Modules .....	10
1.7. New Functions.....	11
1.8. Changed Functions .....	12
1.9. Discarded and Reassigned Functions .....	13
1.10. Opening REAKTOR 3 Ensembles.....	13
<b>2. REAKTOR 5 as Plug-in .....</b>	<b>14</b>
2.1. Automation ID editing .....	15
2.2. Total Recall .....	15
<b>3. Open Sound Control (OSC).....</b>	<b>19</b>
3.1. Application areas.....	19
3.2. OSC System Setup .....	20
<b>4. First Steps in REAKTOR .....</b>	<b>23</b>
4.1. Opening and Playing Examples.....	23
4.2. Your First DIY Synthesizer .....	33
4.3. Your First DIY Structure .....	44
<b>5. Basic Operation .....</b>	<b>50</b>
5.1. Mouse.....	50
5.2. Context Menus.....	51
5.3. Key Commands.....	51
5.4 Ensemble Panel and Structure Windows.....	51
<b>6. Menus .....</b>	<b>53</b>
6.1. File Menu .....	53
6.2. Edit Menu .....	55
6.3. Settings Menu .....	56
6.4. System Menu.....	58
6.5. View Menu.....	65
6.6. ? Menu.....	72
<b>7. REAKTOR Toolbars.....</b>	<b>73</b>
7.1. Main Toolbar .....	73
7.2. Ensemble Panel Toolbar .....	75
7.3. Structure Toolbar .....	76

<b>8. The Browser .....</b>	<b>77</b>
8.1. Accessing Files .....	78
8.2. Auditioning Files .....	80
<b>9. Ensemble.....</b>	<b>81</b>
9.1. Ensemble Structure Window.....	83
9.2. Ensemble Panel Window .....	85
9.3. Ensemble Properties Dialog .....	86
9.3.1. Function Page .....	87
9.3.2. Info Page.....	90
9.3.3. Appearance Page .....	90
9.3.4. Connection Page .....	92
<b>10. Instruments.....</b>	<b>94</b>
10.1. Adding Instruments to an Ensemble.....	94
10.2. Ports.....	95
10.3. Context Menu .....	95
10.4. Instrument Header.....	96
10.5. Instrument Properties .....	98
10.5.1. Function Page .....	99
10.5.2. Info Page.....	104
10.5.3. Appearance Page.....	105
10.5.4. Connection Page.....	108
<b>11. Primary Macros.....</b>	<b>111</b>
11.1. What is a Primary Macro?.....	111
11.2. Adding Macros to a Structure .....	112
11.3. Ports.....	113
11.4. Context Menu.....	114
11.5. Macro Properties .....	115
11.5.1. Function Page.....	115
11.5.2. Info Page .....	117
11.5.3. Appearance Page.....	118
<b>12. Primary Structures .....</b>	<b>120</b>
12.1. What is a Primary Structure? .....	120
12.2. Modules .....	121
12.3. Source Modules .....	125
12.4. Switches.....	128
12.5. Terminals.....	128
12.6. Wires .....	129
12.7. Signal Processing in REAKTOR .....	132
12.8. Context Menu .....	136
<b>13. Panel Editing.....</b>	<b>138</b>
13.1. What Is a Panel?.....	138
13.2. What are Panel Controls?.....	139

13.3. Panel Controls.....	139
13.4. Panel Control Skins .....	147
13.5. Connection Properties of Panel Controls.....	152
13.6. Editing the Panels .....	155
<b>14. Panel Operation .....</b>	<b>156</b>
14.1. Mouse Control .....	156
14.2. Using Keys to Change Control Settings.....	160
14.3. MIDI Control .....	160
14.4. MIDI Out .....	162
14.5. Customized Panels.....	162
<b>15. Snapshots .....</b>	<b>168</b>
15.1. Control ID Numbers .....	169
15.2. Recalling Snapshots.....	169
15.3. Linking Snapshots .....	170
15.4. Managing Snapshots .....	170
15.5. Renaming and Copying Snapshots .....	172
15.6. Comparing Snapshots .....	173
15.7. Snap Isolate .....	174
15.8. Banks Menu .....	174
15.9. Randomizing Snapshots.....	175
15.10. Morphing between Snapshots.....	176
<b>16. Sampling and Resynthesis.....</b>	<b>178</b>
16.1. Sample Management.....	178
16.2. Sample Maps .....	181
16.3. Sample Map Editor .....	184
16.4. Akai Import.....	194
<b>17. Table Modules.....</b>	<b>196</b>
17.1. Properties .....	196
17.2. Context Menu .....	204
17.3. Advanced Operation .....	207
<b>18. “Classic Modular” Macro Collection.....</b>	<b>208</b>
18.1. Display .....	209
18.2. MIDI .....	210
18.3. Mixer/Amp .....	211
18.4. Oscillator .....	213
18.5. Sampler .....	214
18.6. Sequencer .....	216
18.7. LFO, Envelope .....	221
18.8. Filter.....	222
18.9. Delay .....	224
18.10. Audio Modifier.....	225
18.11. Event Processing.....	227

# Content Instrument Guide

<b>Synthesizer .....</b>	<b>230</b>
Carbon 2.....	230
Oki Computer 2 .....	241
SteamPipe 2.....	248
SubHarmonic.....	257
<b>Grooveboxes .....</b>	<b>261</b>
Aerobic .....	261
Massive.....	268
Newscool .....	278
Sinebeats 2.....	283
<b>Sound Generators .....</b>	<b>291</b>
Skrewell .....	291
SpaceDrone.....	294
<b>Sample Player .....</b>	<b>297</b>
BeatSlicer 2 .....	297
Memory Drum 2.....	303
<b>Sample Transformer .....</b>	<b>310</b>
L3 .....	310
Random Step Shifter .....	314
Splitter .....	318
Vectory .....	323
<b>Effects.....</b>	<b>329</b>
FlatBlaster 2.....	329
Lurker.....	332
Space Master 2 .....	339
<b>Sequencer.....</b>	<b>341</b>
SQ16.....	341
SQ8.....	342
SQ 8x8.....	345
SQP.....	347
<b>Index.....</b>	<b>350</b>

# 1. Introduction

## 1.1. What is REAKTOR?

REAKTOR is a powerful and flexible program that turns your computer into a professional-strength synthesis, sampling, and audio-processing system. With REAKTOR's completely modular structure, you can build virtually any digital audio device that you can imagine. From relatively simple analog synths to large modular systems, from basic sample players to exotic granular (re)samplers, from elementary delay lines to full-featured reverb units, your creativity will have virtually no limits.

If building your own instruments and effects is not your top priority, you'll still find plenty to do with REAKTOR. It comes packed with hundreds of instruments and effects of all kinds. Want a simple FM synth? It's there. Want a sample player with independent control of time and pitch shifting? Load it up. Want a multi-effects box to munge your audio files? It's at your fingertips. And the best part of the REAKTOR library is that it enables you to get right down to the business of making music.

If something in the library doesn't do exactly what you need, its modular structure and its control elements are accessible for you to modify. Nothing is hidden. And there's an active user community and online library with new instruments and effects being added all the time. In short, you decide how to use REAKTOR. Fire up a pre-built ensemble (combination of instruments) today, add some snapshots (presets) and make some modifications tomorrow, build your own instrument from the ground up the next day. Just get started!

## 1.2. New/Changed Features in REAKTOR 5

REAKTOR 5 represents a major advancement in flexibility, power, and sonic potential over REAKTOR 4. The following sections present a short overview of new and changed features in REAKTOR 5.

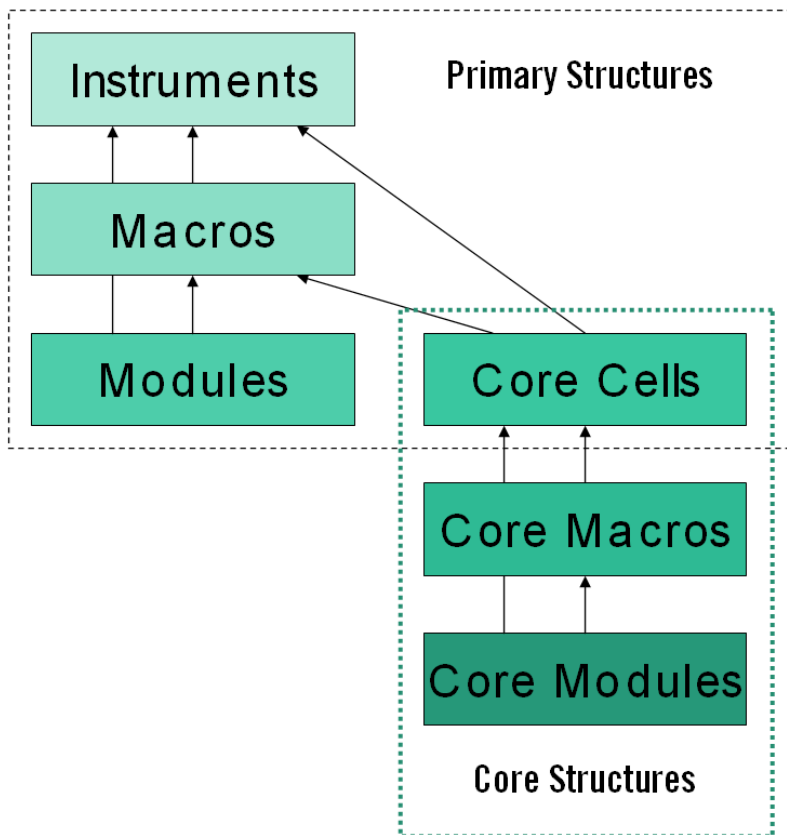
## 1.3. Event Initialization

REAKTOR 5 has a new initialization scheme for event inputs that is used if the REAKTOR 4 Legacy Mode option is disabled (in the Ensemble Properties dialog). We strongly recommend that you disable REAKTOR 4 Legacy Mode in your ensembles for the sake of future compatibility!

## 1.4. REAKTOR Core Technology

The biggest change to REAKTOR 5 is that it provides two levels of functionality: the primary level and the core level.

The primary level comprises the instruments, macros, and modules as they are known from REAKTOR 4.



The core level, also known as the REAKTOR Core, comprises three new objects: core cells, core macros, and core modules. A core cell (\*.rcc file) is a macro/module hybrid that serves as a bridge between the primary and core levels of REAKTOR. Just as primary-level structures consist of primary macros (\*.mdl files) and primary modules, core-cell structures consist of core macros (\*.rcm files) and core modules.



Note that primary and core macros are stored in external files (\*.mdl and \*.rcm, respectively), but that primary and core modules are built into the REAKTOR program. For this reason, modules are referred to as built-in modules.

Core cells, and the core macros and modules they contain, are built upon new concepts of signal propagation and runtime compilation. Using REAKTOR Core technology enables builders to design sophisticated signal-processing structures, that would not have been possible in REAKTOR 4. For a comprehensive introduction to REAKTOR Core technology, please refer to the separate REAKTOR Core manual.

## 1.5. New Primary Modules

There are no new primary modules for audio generation and processing in REAKTOR 5, because this type of low-level functionality is, from now on, realized at the core level (i.e. within core cells), not at the primary level. There is a comprehensive, ever-growing library of core cells, core macros, and core modules for low-level DSP processing.

The new primary modules focus on the user interface, data storage, voice routing, MIDI input/output, and internal connections.

The new modules are:

- **Mouse Area** (Panel) - enables other modules (such as Multi Display and Poly Display) to process mouse actions (button clicks, mouse drags, changes in position, etc.).
- **Multi Display** and **Poly Display** (Panel) - enable REAKTOR users to generate and manipulate multiple graphical objects (rectangles, pictures, animations, etc.).
- **Stacked Macro** and **Panel Index** (Panel) - enable multiple macros to share the same display area in an instrument panel, where one macro is displayed at a time.
- **Channel Message** (MIDI In) and **Channel Message** (MIDI Out) - receive/send all types of MIDI channel messages from/to external MIDI devices (keyboard, sequencer, file, etc.) or internal instruments.
- **Voice Shift** (Auxiliary) - shifts specified input voices (e.g. 1, 2) to specified output voices (e.g. 3, 4).
- **Snap Value Array** (Auxiliary) - stores/recalls arrays of values to/from the edit buffer and snapshots.

- **IC Send** (Terminal) and **IC Receive** (Terminal) - send/receive monophonic event signals anywhere in the ensemble. IC stands for internal connection.
- **Numeric Readout** – is a panel element to display numeric values.

For detailed information on each of these modules, see the **Primary Modules Reference**.

## 1.6. Changed Primary Modules

The appearance and functionality of several REAKTOR 4 modules has been changed in REAKTOR 5:

- **Invert, Rectify** (Math), and **Merge, Order, Value, Logic AND, Logic OR, Logic XOR, Logic NOT** (Event Processing) - the structure icons for all of these modules are different from those in REAKTOR 4.
- **Meter, Lamp, Multi Picture, Multi Text** (Panel), **MIDI In Controller, MIDI Out Controller** - the Internal Connections list in the Properties dialog has been removed from all these modules; Internal connections are now established by the IC Send and IC Receive modules.
- **Snap Value** (Auxiliary) - can now be run in monophonic or polyphonic mode. (In REAKTOR 4, **Snap Value** is a monophonic-only module.)
- **Panel Controls** (Panel) - the functionality of several REAKTOR 4 panel control modules has been changed in REAKTOR 5. Control and port labels can be edited in panel view (in unlocked mode). Control values can be set in panel view (in locked mode). Most panel controls can have panel skins. There are new options for instrument and primary macro background pictures. For detailed information on each of these modules, see the **Primary Modules Reference**.

## 1.7. New Functions

There are several new functions in REAKTOR 5:

- **Panelsets** - an enhanced replacement for REAKTOR 4 screensets.
- **Bookmarking a structure** - you can bookmark a structure so that you can jump straight to it from any other structure in the ensemble.
- **Locking an instrument's voice allocation settings** - an instrument's voice allocation settings (Voices, Max Unison V, and Min Unison V) can now be locked by turning on the Lock Voices option (in the Properties dialog).
- **Voice & MIDI Slave option** - an instrument's voice allocation and MIDI In settings can now be controlled from another instrument in the ensemble.
- **Panel skins** - REAKTOR 5 enables you to customize the appearance of several panel controls by applying skins to them: faders, knobs, buttons, lists, switches, Receive modules, lamps, and meters.
- **Instrument and macro borders** - you can now add borders (blank margins) to instrument panels and framed primary macros.
- **Auditioning audio files in the Browser and Sample Map Editor** - the REAKTOR 5 Browser and Sample Map Editor both support audio-file auditioning (pre-listening).
- **Initialization** - REAKTOR 5 has a new initialization scheme for event inputs that is used if the **REAKTOR 4 Legacy Mode** option is disabled (in the Ensemble Properties dialog).
- **User Content folders** - during installation, REAKTOR 5 creates separate folders for its system files (ensembles, instruments, primary macros, core cells, core macros), and for user files that are created/maintained by the user (ensembles, instruments, primary macros, core cells, core macros, audio, imported files, pictures, snapshots, tables).
- **Deleting wires** - wires can now be deleted by dragging the mouse from the input port to which the wire is connected to a blank part of the structure.
- **Debug option - Show Event Initialization Order** numbers modules in a structure to show their initialization sort order.
- **CPU peak meter** – The CPU meter has been extended. It now also features a bar to show the average CPU drain (white), peak above average (yellow), CPU overload (red).

## 1.8. Changed Functions

Several REAKTOR 4 functions have been changed in REAKTOR 5:

- **Ensemble Panel window** - there is now only one panel window, the former Ensemble Panel window. All instrument panels reside within the Ensemble Panel window.
- **Structure windows** - in order to minimize Structure window clutter, REAKTOR 5 displays all structures (ensemble, instrument, primary macro, core cell, and core macro) in the same Structure window. You can bypass this feature and open a structure in a separate window by Alt+double-clicking the structure icon, or WindowsXP: Right-clicking / OS X: Ctrl+clicking the icon and selecting **Structure Window** from the context menu.
- **Main toolbar** - several aspects of the REAKTOR 4 Main toolbar have been changed in REAKTOR 5. The number of Main toolbar elements has been reduced, because the Ensemble Panel window and Structure windows now have their own toolbars. In the OS X implementation of REAKTOR 5, the toolbar is now displayed as a toolbox that can be placed anywhere on the screen, in order to keep the window headers visible.

There are now two MIDI activity lamps: External MIDI In and External MIDI Out. During the compilation of a core structure, the CPU load indicator changes to a compilation progress bar.

- **Ensemble Panel and Structure toolbars** - Ensemble Panel window and Structure windows now have their own toolbars, each with a set of the most commonly used functions in that window.
- **Instrument header** - Several aspects of the REAKTOR 4 Instrument header have been changed in REAKTOR 5. The A, B, and Minimize buttons have been moved to the left. The panel Lock/Unlock function now has its own button (wrench icon). The Mute and Solo buttons have been removed. There are now four MIDI activity lamps: External and Internal MIDI In, and External and Internal MIDI Out. The In and Out drop-down menus provide access to all of the input and output connections (MIDI and wiring) of the instrument.
- **Browser item access** - The REAKTOR 5 Browser provides dedicated buttons that enable you to fast access system and custom folders.

## 1.9. Discarded and Reassigned Functions

Several REAKTOR 4 functions have been discarded or reassigned in REAKTOR 5:

- There are no longer separate panels for instruments. All instrument panels are displayed in the Ensemble Panel window. The new Panelset bar provides easy (one-click) access to all of an ensemble's instrument panels.
- REAKTOR 4 screensets (storage slots for ensemble layouts) have been replaced by REAKTOR 5 panelsets (see **New Functions, panelsets**).
- The Browser no longer supports wiring (this has been reassigned to the In and Out menus in an instrument's panel header), structure browsing, and module loading.
- The internal MIDI connections of an instrument are no longer set in the instrument's Properties dialog; they are set in the instrument panel header's In and Out menus.

## 1.10. Opening REAKTOR 3 Ensembles

Ensembles saved with REAKTOR 3 will not open in REAKTOR 5 unless the REAKTOR 3 USB copy protection key is plugged in. If you have the key, install it and plug it into your USB port, then open the REAKTOR 3 ensembles in REAKTOR 5 and save them as REAKTOR 5 files. Once you've done this, you'll be able to open the ensemble files without using the key. There is also a Batch Processing function to perform the conversion of many files at once.

## 2. REAKTOR 5 as Plug-in

The plug-in version of REAKTOR looks a bit different from the standalone version, but you have still access to all the software's main features (unless they are not applicable to plug-in operation).

A menu is available in the plug-in version as context menu. To call up the menu, WindowsXP: Right-click / OS X: Ctrl + Click within an empty space of the toolbar. You can hide the toolbar with the first entry in this context menu. Even if the toolbar is hidden, you can call up the menu anytime with a WindowsXP: Right-click / OS X: Ctrl + Click on an empty area of the Browser, Snapshot or Properties window.

If you hide the toolbar and close all other windows except for the Ensemble window and finally press the Resize button, your Ensemble fits perfectly into the plug-in window.

The left part of the plug-in window can be toggled between three different views: Properties (F4), Browser (F5), Snapshots (F6). Alternatively, you can hide this area by clicking on the Close button which shows a small cross.

If the toolbar is visible, the button on the right serves for switching to the **Browser** view. If it is hidden, use the shortcut F5.



Browser button

The **Snapshot** view can be accessed by the Snapshot buttons in the Ensemble and Instrument Headers (F6).



Snapshot button

The **Properties** view can be accessed by double clicking on any control within an Instrument panel (F6).

If you open the **Sample Map Editor** for a Sampler by double clicking a sampler waveform within an Instrument panel, it will be displayed at the bottom of the plug-in window.

Ensembles can be saved using the **Save** button in the Toolbar. You can save the Ensemble under a new name when you perform a **Ctrl** click on the **Save** button.

**Automation:** If your host supports plug-in automation, REAKTOR will pass the parameter names and value ranges of the controls used in the currently loaded ensemble to the host.

## 2.1. Automation ID editing

Each control element in a REAKTOR ensemble is assigned a unique automation ID. This enables automation from a host application. The ID determines the order that the controls appear in the host parameter list. For this reason, the instrument properties has several functions for re-arranging the automation ID's. These functions are particularly important in certain host applications that only recognize a limited number of parameters.

- Compress ensures that there are no 'gaps' between automation ID's.
- Sort and compress additionally ensures that ID's of controls within the same macro are grouped together (so that they will appear together in the host parameter list).
- Each instrument in an ensemble also has a unique base ID. This determines the overall order of Instrument controls within the parameter list. Instrument up and Instrument down increase or decrease the priority of the instrument in the parameter list.

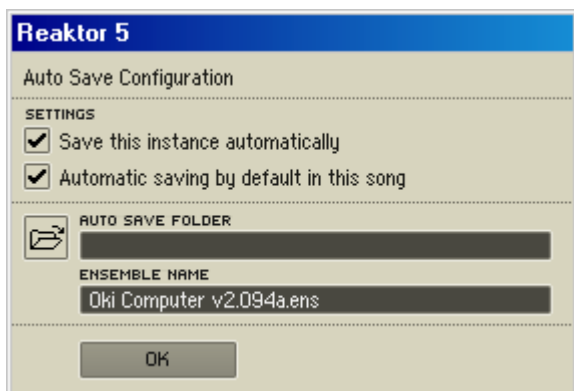
## 2.2. Total Recall

### Loading a REAKTOR Plug-in

After inserting a REAKTOR plug-in in your host application, you will be presented with two options: **New Ensemble** and **Load Ensemble....**

Selecting **New Ensemble** will load **New.ens** (an empty structure in which you can construct your own ensemble). Selecting **Load Ensemble...** will present a file dialog to select an existing file.

It is also possible to load an ensemble by drag and drop from the browser or from an outside window.



The Auto Save Configuration Dialog

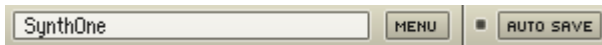
After loading an ensemble or creating a new one, the **Auto Save Configuration** dialog with the following options will appear.

- **Save this instance automatically:** Enabling this option is always recommended. It ensures that the ensemble and all changes made to it are saved to a separate .ens file when you save your project. This is safest as amendments to the original ensemble will not affect your project. When enabled, you are required to choose a destination folder and a filename for the Auto Save location. When disabled, a copy of the ensemble is not made, thus saving disk space. The current setting of the ensemble panel controls will be stored within your project file. But please note, sharing ensembles in this way means that amendments to the structure of the original ensemble will affect all projects using the ensemble. Further, changes to event / audio table data and sampler file references will not be stored when saving the project.
- **Automatic saving by default in this project:** When this option is on, the Auto Save Configuration dialog will open up with each REAKTOR plugin that you insert in this project. Keeping this option active guarantees that the latest state of all ensembles used in this project will be saved in their own files.
- **Auto Save Folder:** Here you specify the folder where the ensemble shall be automatically saved. You can either enter a directory from the keyboard, or select it from a file browser by clicking on the folder icon. It is recommended to store ensemble files in the same folder as the project file itself. This ensures that they will be kept together when moving the project folder.



- **Ensemble Name:** Here you can specify a filename for the ensemble. We recommend choosing a name that is unique and cannot be confused with the names used for other plug-in instances in the current project or other projects.
- **OK:** Selecting OK confirms the Auto Save configuration. If Auto Save is on, but a folder or ensemble name have not yet been defined, the option to select “OK” is disabled.
- **Cancel:** Discards changes and closes the dialog

## Auto Save Functions in the Plug-in Header



- The first field shows the file name of the currently loaded ensemble. This is updated with the file name that has been given in the Auto Save Dialog.
- The **Menu** button opens the main menu.
- The lamp between Menu and Auto Save” buttons shows whether the plug-in is in Auto Save mode or not.
- The **Auto Save** button opens the Auto Save Configuration dialog. This allows you to reconfigure Auto Save settings. If you change the Auto Save folder, the ensemble will be automatically moved to the new destination. If there are any other instances using the old folder, you will be asked whether to change the Auto Save folder for those instances too.

## Replacing an Ensemble

If Auto Save is enabled, and you replace the ensemble, the Auto Save configuration dialog will appear. Thus, when auditioning different ensembles it might be advantageous to disable Auto Save until you decide which ensemble to use.

## Loading a Project and the Ensemble is not Found

If an ensemble is not found when opening a project, a message will appear in the plug-in window. You can locate the ensemble via a file browser by clicking the **Locate Ensemble** button, by dragging the correct ensemble into the window, or by re-configuring the Auto Save dialog. If other plug-in instances have missing ensembles, an option to try the new folder location for those other instances will now appear.

## Working with multiple REAKTOR Plug-ins

If you load another REAKTOR plug-in into your project, the Auto Save mode and folder are taken from the last REAKTOR plug-in instance.

## What is Saved with a Project

When the project is saved (or the host queries the plug-in data for whatever reason), the following happens:

- If Auto Save is turned on, the ensemble is saved as a file to the AutoSave folder.
- If Auto Save is turned off, the path of the currently used ensemble is saved in the project data.

In either case, the following are also saved:

- The current setting of all panel controls
- The last Auto Save mode and folder
- The current window size and mode (minimized, auto-layout or fixed)

## Saving a copy of an Ensemble

In the Plug-in the 'Save As' command is replaced by 'Save A Copy As'. This allows saving a copy of the ensemble elsewhere without changing the Auto Save filename and path.

## Global Auto Save default

In the Preferences, the **Auto Save off by default** option determines the default Auto Save configuration for new instances.

## Plug-in Size Functions



Several functions are available for controlling the window size of a plug-in.

These are the last four buttons on the right hand side of the plug-in toolbar. The first two buttons minimize and maximize the screen respectively. The third button ('Manual resize') resizes the plug-in window to fit the ensemble when pressed. The fourth button ('Automatic Resize') is a toggle button. When activated, the plug-in window will always be automatically resized when required (for example, switching windows, toggling between A and B panels).

The maximum plug-in window size can now be configured in the preferences.

## 3. Open Sound Control (OSC)

OSC is an open, network-independent protocol developed for communication among computers, sound synthesizers, and other multimedia devices. Compared to MIDI, OSC provides increased reliability, greater user convenience, and more reactive musical control. Open Sound Control is useful in any situation where multiple music applications have to work together on the same computer or on networked computers. While MIDI only has the parameters defined in the standard (note on/off, pitch bend, control change, etc.), OSC lets each program have its own symbolic, hierarchical, and dynamic address space.

OSC can be used with any networking technology, including TCP/IP based LANs and the internet. OSC's time tags and bundles of messages provide for exact timing of musical results even if the network has latency and jitter. OSC supports a variety of argument types which will be successively integrated into future releases of REAKTOR.

### 3.1. Application areas

The OSC implementation of REAKTOR allows for easy setup of

- Internet-based collaborative international music making
- Sound installations with dozens of computers coordinating with each other
- Coordinating synthesis between two (or more) computers to increase the total processing power
- Communication between music software applications within a single computer.

The OSC implementation in the current version of REAKTOR only supports transmission of event data between two or more REAKTOR computers, but not audio data. In addition to REAKTOR's general requirements you will need an ethernet card to use OSC. Also, TCP/IP and UDP protocol stacks must be installed on your computer.

## 3.2. OSC System Setup

**OSC Setup**

**OSC**  
☒ Activate  
LOCAL IP ADDRESS: 127.0.0.1  
LOCAL PORT: 10000  
LOCAL IDENTIFIER: Reaktor5-1  
Apply

**CLOCK SYNC**  
☐ Master ☒ Off  
SELECT MASTER: Off

**TIME SYNC**  
☐ Master ☒ ok  
SYNC MESSAGES: ok  
NETWORK DLY (MS): 0.000  
TIME OFFSET (MS): 0

Identifier	IP Address	Port
------------	------------	------

Scan  
Edit  
Delete

IDENTIFIER: REMOTE IP ADDR.: REMOTE PORT: 0  
Apply

**OSC MESSAGE**

MONITOR OPTIONS: select

**OSC MONITOR**

OK Cancel

OSC Settings window

REAKTOR's Open Sound Control (OSC) settings are made using the **OSC Settings** window which you can open from the **System** menu. OSC provides communication between media devices and software such as REAKTOR using a variety of network protocols, including TCP/IP and LANs.

## Activating OSC

OSC communication can be enabled and disabled at will using the **Activate OSC** button at the top-left of the OSC Settings window. OSC communication is only possible when REAKTOR's audio processing is active, and as a consequence, you need an audio card or built-in audio capability on your computer to activate OSC. The Activate OSC status is preserved between REAKTOR sessions.

## OSC Identification

In addition to the Activate OSC button, the top section of the OSC Settings window contains your Local IP Address, Local Identifier, and Local Port settings. The settings in this section are all preserved between REAKTOR sessions.

- **Local IP Address:** This is the current IP address of your computer. It is recognized automatically and can not be edited.
- **Local Identifier:** This name will be used to identify you to other OSC clients. You can choose any name you like.
- **Local Port:** This is the sub-network identifier by which other OSC clients recognize your system when they scan the network (see the Scan button below). Only certain ports are scanned, and you should use a number between 10,000 and 10,015.
- **Apply:** When you make changes, you need to click the Apply button to have them take effect.

## OSC Synchronization

The second section of the OSC Settings window contains synchronization settings.

- **Clock Sync (Master):** Click this to have REAKTOR send an OSC clock signal to other OSC clients. OSC clock works exactly like MIDI clock. Clock will be sent to all clients on the Member list (see below).
- **Time Sync (Master):** Time Sync is a control circuit system. The client constantly polls the master for the time stamp, compares the received time with its own, and adjusts it if necessary.
- **Select Master:** When not operating in Clock Sync Master mode, use this menu to synchronize to an OSC master. Select Clock Sync to synchronize to Clock Sync signals. Select another OSC member to Time Sync with that client.
- **Sync LEDs:** There are small LEDs to the right of the Clock Sync and Time Sync checkboxes. These indicate when a synchronization signal is received or sent.

- **Sync Errors:** This field reports synchronization errors.
- **Time Offset (ms):** Adds the time offset to each OSC message sent to the clients. If you enter 1000 ms each message will be received one second later by the client. This only applies if the participating clients are in Time Sync mode.

## OSC Member List

This list contains all REAKTOR OSC clients to whom a connection has been established.

You can edit and delete entries in this list. To do so, select an entry and press the **Edit** button. To apply any changes you have made, click the **Apply** button.

To delete an OSC connection in the OSC member list, just select the entry and press the **Delete** button.

The **Scan** function is able to recognize OSC members within a sub-network automatically. This only works when the following conditions are met:

- The client must be located within the same subnet.
- REAKTOR must be running on this computer (audio engine active).
- OSC has to be activated in the REAKTOR OSC Settings.
- In the REAKTOR OSC Settings a port address between 10,000 and 10,015 must have been entered.

If you want to connect two computers that are not located in the same subnet (for instance if you want to establish an OSC connection via the internet), you will have to enter manually the **Identifier**, **IP address** and **Port number** of the other computer below the member list area and then press the **Apply** button.

## OSC Monitor

The bottom section of the OSC Settings window is for monitoring OSC activity.

- **OSC Message:** This field is for sending text messages to other OSC clients. It can be used to test OSC connections or as a chat box. First select a recipient in the Member list, type a message and finish the operation with the enter key. The message will then be sent to that client.
- **OSC Monitor:** The monitor displays all received OSC messages.
- **Monitor Options:** Here you can set certain functions for the monitor window.

## 4. First Steps in REAKTOR

The purpose of this chapter is to make you familiar with the basics of the operation and the functionality of REAKTOR and how to program it.

We will dispense with trying to tell you that REAKTOR is a very simple affair and that within only a few minutes you will have programmed your own physical modeling synthesizer. That would be a lie. The fact is that REAKTOR is a complex program that offers complex functions which allow you to achieve complex things. And if that's just what you want to do, you won't really get around an intensive initial learning phase. After all, real success never comes easy.

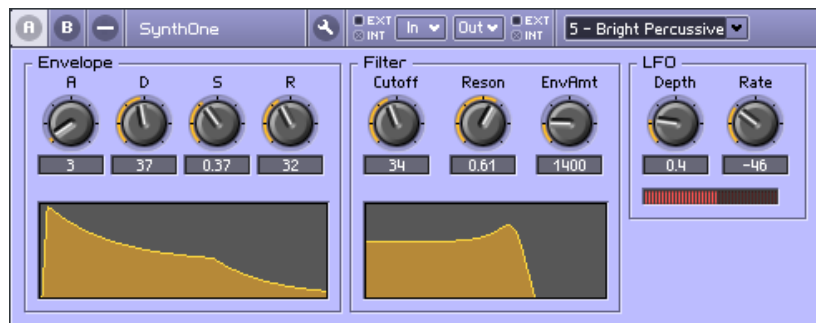
But don't worry. Although you can work with REAKTOR at a complex level, you don't have to. As you will see in our first tour, it is possible to make music with the software using a number of different instruments, even without any knowledge of synthesis methods or processing structures. You simply help yourself to the provided library.

### 4.1. Opening and Playing Examples

First, make sure that your MIDI controller instrument (master keyboard or MIDI workstation) is connected to one of the MIDI inputs of your computer. The input port should have been activated under **Input Interface** on the **MIDI** tab of the **Audio + MIDI Settings...** dialog. (For more information about activating MIDI ports, see **REAKTOR Standalone**). The MIDI transmit channel on your controller should be set to 1.

Alternatively, you can just use the QWERTY keys on your computer keyboard to play notes (see **Appendix** for key mapping).

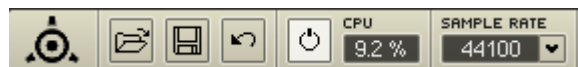
## SynthOne



Panel Window for SynthOne

Now open the folder **Tutorial Ensembles** in the REAKTOR Library folder, select **SynthOne.ens** and click on **Open**, or drag the ensemble from the REAKTOR Browser into the main REAKTOR window.

You should first take a quick look at the Main toolbar.



Is the **Run/Stop Audio** button on, and does the field next to it show the current CPU usage? If so, you can now play **SynthOne**. If not then click the **Run/Stop Audio** button to turn on your new synth, and then let 'er rip! By the way, if the CPU usage field shows **Over** or a warning message pops up (“Processor Overload!”) to inform you that audio processing has been turned off, then you will need either to reduce the number of **Voices** in the Instrument header of the Synth 1 panel from **6** to a smaller number, or to choose a lower sample rate than the initial 44100 Hz in the toolbar. And if the **Out** level meter should ever light up red, this indicates that the sound card is being overloaded – in which case you should reduce the ensemble volume (by using the Main fader).

With every note you play, the MIDI In lamp in the Instrument Header should light up red.



This lamp indicates that the SynthOne is receiving MIDI data, while the MIDI In Ext lamp in the Instrument header indicates that the MIDI data is being received by the instrument.



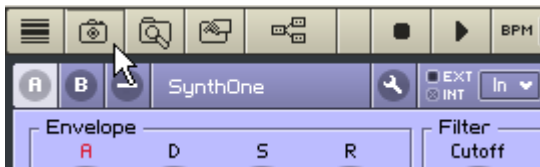
**SynthOne** is a replica of a simple 6 voice analog synthesizer. It contains one sawtooth oscillator, a 24-dB lowpass filter, an LFO that affects the pitch, and an ADSR envelope that modulates the sawtooth amplitude and the filter cutoff frequency.

In the **SynthOne** panel you can find the control elements that are available for changing the sound. From left to right, these are: **A**(ttack), **D**(ecay), **S**(ustain) and **R**(elease) for changing the shape of the envelope, **Cutoff** for controlling the filter cutoff (or corner) frequency, **Reson**(ance) for the amount of boost applied to the frequencies near this cutoff frequency, **EnvAmt** for setting how much the filter is affected (modulated) by the envelope, and finally **Depth** for setting the LFO intensity and **Rate** for the LFO speed.

If you have ever had your hands on a synthesizer before, then dealing with this straightforward device should not be much of a challenge. On the other hand, if **SynthOne** is your first synth, you now have the opportunity to experiment with the effect that these few but essential synthesis parameters have on the sound. And there's no reason to worry about getting too lost.

If during your experiments you should come across a sound that you particularly like, you may want to save it. To do so:

- Open the **Snapshots** window by clicking on the camera icon in the toolbar, which opens the snapshot window.



The **Snapshots** window is used for managing (saving, renaming, deleting, etc.) snapshots, which are like the “patches”, “presets” or “programs” found in other programmable synthesizers.

- Click the **Append** button twice (the first click lights the button, the second un-lights it) to save your current **SynthOne** sound settings to the first empty slot in the snapshots list.
- REAKTOR gives your new snapshot a default name. To rename it, double-click on the current name, type your new name, and press the **Enter** key.

## Padecho



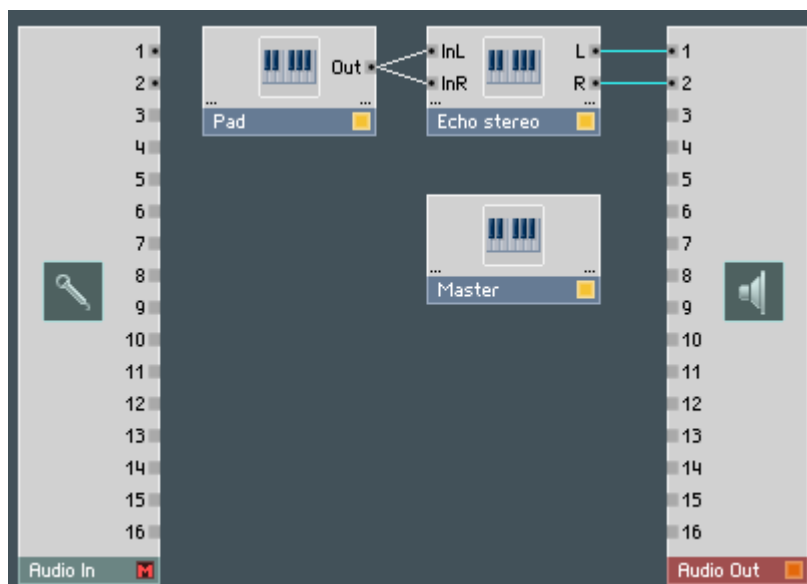
The next ensemble we are going to have a look at is called **Padecho.ens**. (Pad-echo = Pad with echo.) You will find it in the same folder from where we previously pulled out **SynthOne**.

At this point you will be asked whether you want to save the changes you have made in SynthOne. You probably want to answer **No** here, unless you have just created a snapshot(s) worth keeping.

You will now see three instrument panels, which are labeled Pad, Echo Stereo and **Master**. Click on the Structure button (its icon is three little boxes connected by wires) in the Panel toolbar in order to open the Ensemble Structure window.



The ensemble is the highest level in REAKTOR and the Ensemble Structure window provides a bird's eye view of the complete working environment that is available to you. In this case it contains the synthesizer **Pad**, the stereo delay effect **Echo Stereo**, and **Master** which holds the master ensemble controls **Main** and **Tune**.



The output of **Pad** is connected to the two inputs of **Echo Stereo**. The outputs of **Echo Stereo** are then connected to the top two inputs of the **Audio Out** module.

You will find this **Audio Out** module in every ensemble. It represents the software's connection to the rest of the world, which is normally the audio output of your sound card, but it can also be the Plug-In connection to another piece of software. Its counterpart, which is also present in every ensemble, is the **Audio In** module, which represents the audio inputs of your sound card (or the Plug-In connection). In this case **Audio In** is muted (as indicated by the red **M** in its title bar), because **Padecho** doesn't use any audio input.

A quick look at the **Padecho** ensemble already brings to light two essential features of REAKTOR. One is that an ensemble can consist of more than one instrument. The other that its generative power is not restricted to synthesizers, because **Echo Stereo** is an effects unit.

You can switch between the Structure and Panel windows by double-clicking the black background of the Panel window, or double-clicking the dark grey background of the Structure window. Give it a try.

You can play the ensemble with the Structure window open (as pressing a key on your MIDI instrument will show), but there are no control elements at your disposal, which does detract from the entertainment value of a synth quite a bit.

The panel **of the Master Instrument** gives access to two knobs: **Main** for controlling the master volume of the ensemble, and **Tune** for setting the master tuning.

Together with the panels for **Pad** and **Stereo Echo** you see all control elements of the ensemble.

Before we give you some time alone with the **Padecho** ensemble, we'll make a few short comments regarding its structure. The **Pad** synth contains two oscillators that both generate a pulse-wave. The tuning of the second oscillator can be controlled relative to the first one, coarsely with the knob labeled **Interval**, and finely with the **Fine** knob. The pulse width of both oscillators is set with **PWidth** and can also be modulated with the LFO. **LFO rate** sets the speed, and **Depth** the amount of LFO modulation. The controllers for the ADSR envelope, which again affects both filter and amplitude, as well as the knobs to the right for controlling the filter, correspond to those we got to know in **SynthOne**.

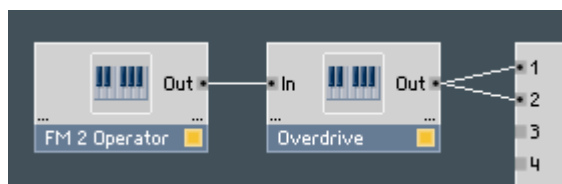
The **Echo Stereo** consists of two delay lines, one of which processes the left and the other the right stereo channel. Their delay times can be controlled independently of each other using **Del L** and **Del R**, where a setting of zero means that there is no delay. The desired number of echo repeats is set with the knobs **F(eed)Back** and **Cross**, where **FBack** controls the amount of signal of a channel (L or R) going back into itself and **Cross** the amount going into the respective other channel. Finally, **Wet-Lvl** sets how much of the original signal goes through the delays, controlling the strength of the effect.

## FM Overdrive



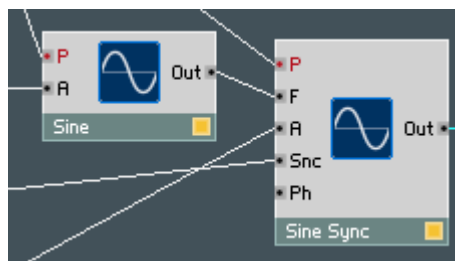
The next ensemble we want to present to you is called **FM-Overdrive.ens**. (**FM-Overdrive** = FM with 2 Operators + Overdrive.) You can also find this in the by now familiar **Tutorial Ensembles** folder.

As a quick look at the Ensemble structure shows, we again have a combination of two instruments here: the synthesizer **FM 2 Operator** followed by the distortion effect **Overdrive**.



The **FM 2 Operator** synthesizer demonstrates the flexibility of REAKTOR. In addition to subtractive synthesis, REAKTOR is capable of other types of synthesis. In this case, FM (frequency modulation), made popular by the Yamaha DX series of synthesizers, is used for tone generation.

In our example there aren't 6 operators (as in the DX7), or 4 like in the smaller DX models, but only 2 operators, so the whole structure should be quite clear.



Both operators consist of an oscillator that generates a sine wave. One, called the carrier, is responsible for generating the fundamental wave and thus setting the pitch of the sound. The other operator is the modulator that affects the frequency of the carrier and controls the sound's timbre.

Play a few notes on your MIDI instrument. Not very exciting, is it? Now slowly turn the **FM** knob upwards and listen to how the sound changes.



A bell-like element starts to creep into the sound until it dominates it completely when the maximum position is reached. On a technical level, all we have done by sliding the **FM** knob up is to increase the level of the modulator and thereby determine how much it modulates the carrier's frequency.

In the next step we turn our attention to the **Interval** knob. The effect that this parameter has should quickly become quite clear. The knob placed next to it, **Detune**, allows you to make fine adjustments to the interval setting.

A very simple envelope is responsible for determining the sound's development over time. The carrier's envelope, which controls volume, has only the two parameters **D**(ecay) and **R**(elease). The envelope for the modulator is even simpler and has only the one knob for setting the decay, labeled **Mod-D**.

Armed with this knowledge you should not find it difficult to create your own sounds with this 2 operator FM synthesizer, and to do it with a sense of purpose.

Let's turn briefly to the **Overdrive**, the purpose of which is simply to furnish your FM sound creation with some amount of acoustic grit. The best thing is probably if you first try out the various snapshots before dedicating yourself to the following explanation of this device.

**Drive** sets the level of the signal that is sent to the distorting element and therefore controls the amount of dirt that is generated. With **Asym** it is possible to modify the overtone spectrum of the signal in such a way as to make it "warmer", i.e. to make it sound as if the sound was generated using a valve (tube) circuit. The distortion circuit is followed by a filter with the parameters **Freq**(ency) for setting its cutoff frequency and **Emph**(asis) to emphasize this frequency. The setting of the **Volume** knob determines the output level of the sound signal.

## 16-Step Sequencer Plus Bassline



The ensemble **Squnc16\*.ens** (Squnc16 = 16-Step Sequencer) with which we are now going to experiment, can also be found in the **Tutorial Ensembles** folder.



A look at the Ensemble structure tells us something about the construction of this ensemble: **Sequencer16**, a 16-Step sequencer (another device that REAKTOR can provide) controls **Baseline**, a kind of 303 clone, whose signal reaches the audio output via the **Auto Panner**.

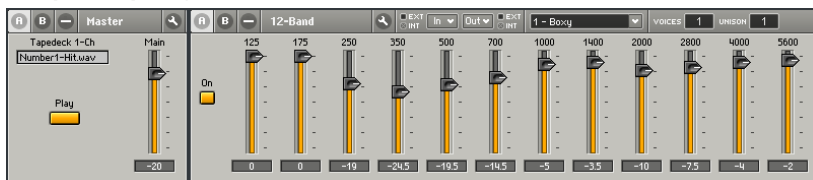
The panel of **Sequencer16** is already visible and, if the **Run** button is already pushed, you should hear the pattern playing. Press **Run** once to stop the pattern and another time to restart it. You can change the pitch for every step with the **Pitch** faders in the top row, and the volume for each step with the **Lvl** (Level) faders in the row below. The tempo is adjustable using the **BPM** knob (next to the **Run** button), and the length of the notes can be manipulated with knob labeled **Length**.

Finally, the **Reset** button located underneath **Run** resets the sequence to step 1 every time it is pressed. If it is pressed while the sequencer is running, a nice shifted pattern can be generated. If it is pressed while the sequencer is stopped, this ensures that, on starting, the sequence will begin at the first step and not somewhere in the middle.

The **Baseline** instrument corresponds to what you may know from a 303; but even if you have never seen such a beast, with such a small number of controls it's unlikely that any confusion will arise. Just turn any knobs you want and listen to the result.

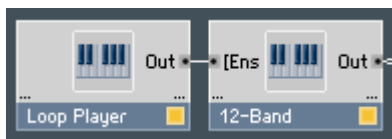
As you have probably already noticed, the sound in this ensemble is always moving back and forth between the left and right speakers. The **Auto Panner** is responsible for this. With **Amount** you set by how much the signal travels between the left and right channel and **Rate** is the speed of this movement. That is all there is to it.

## Sample Loop Player



The final example for illustrating REAKTOR's capabilities is called **Wav-play\*.ens**. You will find it with the previous REAKTOR examples in the **Tutorial Ensembles** folder.





This ensemble is made up of the units **Loop Player** and **12-Band**. Before you can hear anything, you first need to load a sample into **Loop Player**. Windows XP: Right-click / OS X: Ctrl+click on the sample slot displaying “untitled\*.wav” in the panel and choose **Load Audio in Tapedeck...** from the context menu. In the Open Audio File dialog select any one of the WAV or AIF files on your hard drive and load it by clicking on **Open**. Presto, a click on the **Play** button and you should hear the newly loaded sample as a loop. If **Play** is already pressed and you don’t hear anything, or if the sample fails to loop, press **Play** twice to reinitialize it and start playback.

The whole point of the **Wav-play** ensemble, however, is to be found in the **12-Band** effect. The panel looks very much like the classic control panel of a graphic equalizer – that is, faders which allow the level of various frequency bands to be controlled. What we have here is a filterbank which allows even more dramatic manipulation of the sound than an equalizer. Each fader has a number that indicates what frequency band (measured in Hz) it controls. Try out the effect of the different frequency bands on the sound, while the loop is running. You will notice that it’s not just the timbre that changes, but that it is possible to nearly remove entire parts and so manipulate the musical character of the loop.

## 4.2. Your First DIY Synthesizer

As you may have noticed in the previous examples and by further rummaging through the library, REAKTOR offers a wealth of ready-made instruments, effects units and combinations. But the true thrill of REAKTOR is in the possibility of designing and constructing your own instruments. And as you will see, it isn’t difficult if approached in the right way.

How about a good old-fashioned analog synthesizer? Let’s do it using subtractive synthesis, where an oscillator first produces a signal rich in high frequency components, some of which are subsequently removed using a time variable filter. All right, here we go...

### Preparation

To construct our synthesizer we will use a method that is very effective – the use of **macros**.

---

**Note:** REAKTOR supports two kinds of macros: primary macros (macros that reside within the primary level of REAKTOR) and core macros (macros that reside within the core level of REAKTOR). In this section, we are speaking exclusively about primary macros. For DIY information on core macros, see the REAKTOR Core manual.

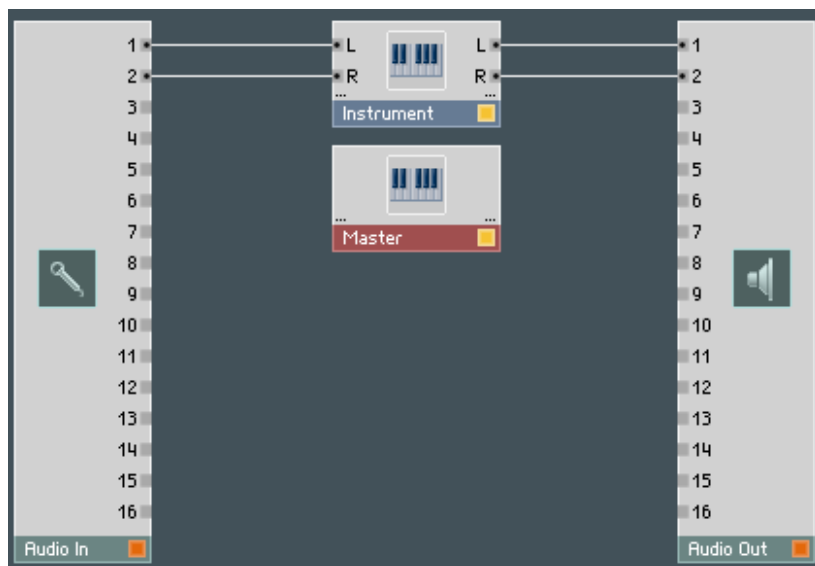
---

In REAKTOR terminology, macros are functional blocks that make the construction of complex structures quite easy, and most importantly, everything remains clearly laid out. In REAKTOR you already have an extensive library of such macros at your disposal, and we will help ourselves to it.

Initially, turn off the **Run/Stop Audio** button in the Main toolbar, so that you don't get startled when the half-finished construction suddenly starts making noises.

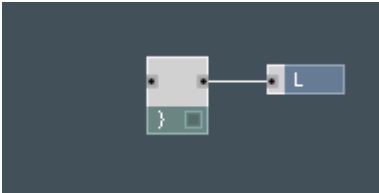


To begin with we will prepare the workspace in which to construct the synth. Please open the **File** menu at the top of REAKTOR and choose the entry **New Ensemble**. After this you will see (in the Ensemble structure) our old friends **Audio Out** and **Audio In**, as well as two instruments, **Instrument** and **Master**.

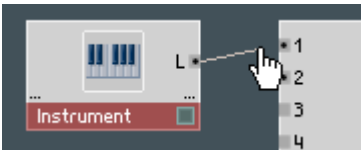


To proceed, delete the default **Instrument** since we want to see how to build an ensemble from ground up. The other instrument called **Master** should remain, since it contains the important global controls **Level** and **Tune** which appear in the Panel window.

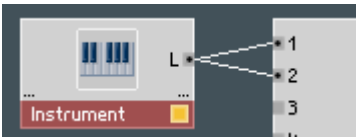
First we need a shell – a box so to speak – in which we will construct our synth. For this we take an empty instrument which we find in the library. XP: Right-click / OS X: Ctrl+click on a blank part of the Ensemble Structure window and in the context menu choose **Insert Instrument** ⇒ **New - 2In2Out**. An empty instrument named **Instrument** appears in the structure. Double-click **Instrument** to open its structure and delete all terminals inside it except for the **L** output terminal and the **Audio Voice Combiner ( }**) in front of it.



Double-click on a blank part of the **Instrument** structure to redisplay the Ensemble structure window. (Double-clicking on a structure is a great shortcut for moving one structure up in the ensemble hierarchy.) Click on the **L** output port of **Instrument**, drag the mouse pointer to **1** input port of the **Audio Out** module, and release the mouse button.



Do you now see a wire connecting the two components? If not, try again. If so, we congratulate you on creating your first virtual wire! In the same way connect the **L** output of **Instrument** to the **2** input of **Audio Out** so that you can later hear the sound on both channels.



## Choice of Components

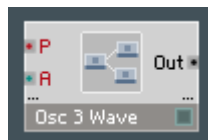
For our synth we need one or more oscillators whose output signal should go through a filter and whose volume should be controlled by an envelope, and that's it. We will now assemble these components.

Take a look at the **Master** and **Instrument** panels (not structures!) in the Ensemble Panel window.



The **Instrument** panel is empty (because we haven't added any visible objects to its structure yet), and **Master** contains only the **Level** and **Tune** controls.

Open the Structure window for **Instrument**. It is here that we will insert the components mentioned above. XP: Right-click / OS X: Ctrl+click in **Instrument's** Structure window and in the context menu choose **Macro** ⇒ **Building Blocks** ⇒ **Oscillators** ⇒ **Osc (pls, saw, tri)**. In the Structure window you can now see the insert macro, which is named **Osc 3 Wave**.



A quick look at the **Instrument** panel, which was empty until just a few moments ago, shows the controls contained in **Osc 3 Wave**.

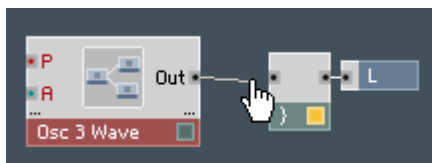


We're making progress.

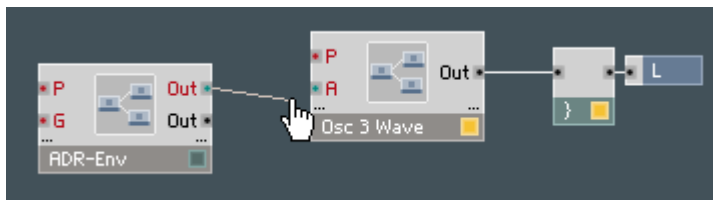
Before we go on we should make sure that the oscillator is working. As a precautionary measure, set the **Level** fader of **Master** to, let's say, **-10** to avoid any nasty surprises during the following audio test.



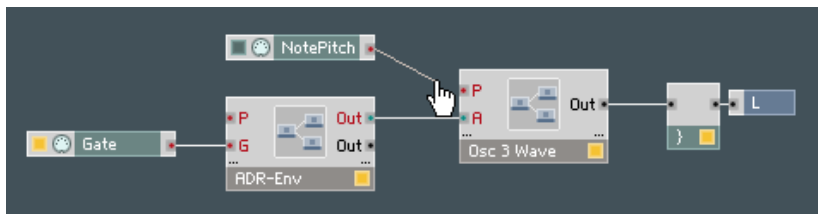
In the **Instrument** structure, connect a wire from the **Out** port of the **Osc 3 Wave** macro to the input port of the **Audio Voice Combiner** module ( **}** ). To do this, click and drag your mouse from one port to the other (the direction doesn't matter). The **Audio Voice Combiner** serves to convert a polyphonic signal into a monophonic one. This conversion is especially important in front of an instrument output port, since instrument ports are generally monophonic.



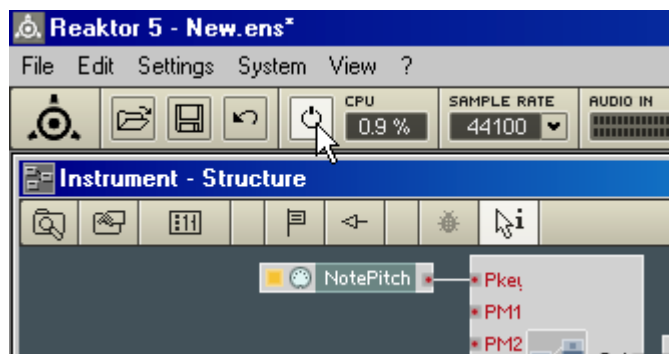
Let's add an **ADSR** volume envelope to the **Osc 3 Wave** macro by using the context menu again and choosing **Macro** ⇒ **Building Blocks** ⇒ **Envelopes** ⇒ **ADSR - Env**. Connect the lower **Out** output port (the black one, not the red one) of the **ADSR-Env** macro to the **A** input of the **Osc 3 Wave** macro.



Now we only need two more important MIDI modules to get a connection to an external MIDI input device. Insert the **NotePitch** module (**Built-In Module** ⇒ **MIDI In** ⇒ **Note Pitch**) and connect it to the **P** input of the **Osc 3 Wave** macro. Finally we need a **Gate** module (**Built-In Module** ⇒ **MIDI In** ⇒ **Gate**), which has to be connected with the **G** input of the **ADSR-Env** macro.



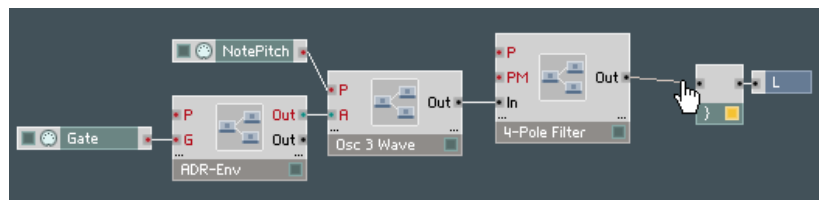
Turn on the **Run/Stop Audio** button in the Main toolbar, and press some keys on your MIDI keyboard (or computer keyboard). You should hear sounds from the synth. (If not, check your wiring, and/or save/reload the ensemble.) This is it, our oscillator in its raw form – not really beautiful (yet) but audible.



Before loading the next component, the filter, first remove the wire you just drew between **Osc 3 Wave** and the **Audio Voice Combiner**. Simply drag from one port to another again, as if connecting a second wire, and the connection is gone. Alternatively, just click on the wire to select it (it changes color) and press the **Del** key.

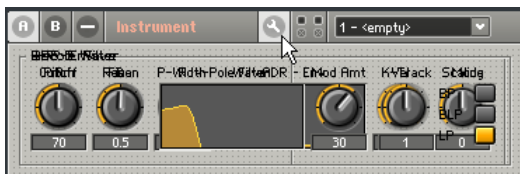


Now load a Filter macro in the same manner as you did the other macros. You will find it in the context menu under **Macro** ⇒ **Building Blocks** ⇒ **Filter** ⇒ **4 Pole Filter (BP, BLP, LP)**. Connect the **Out** port of the **Osc 3 Wave** macro to the **In** port of the **4 Pole Filter** macro, and then connect the **Out** port of the **4 Pole Filter** to the input of the **Audio Voice Combiner** so that you get a signal at the output again.

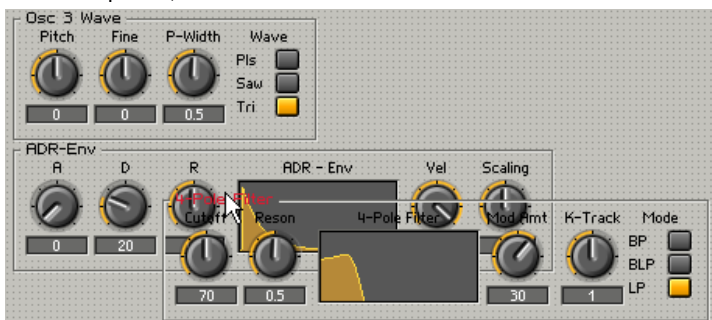


Note that several new filter controls have appeared in the panel. If you cannot see these controls, because your **Instrument** panel is a jumble of overlapping controls, do this to clean it up:

- 1) Click the **Lock/Unlock Panel** button (wrench icon) in the **Instrument** panel header to unlock the panel;



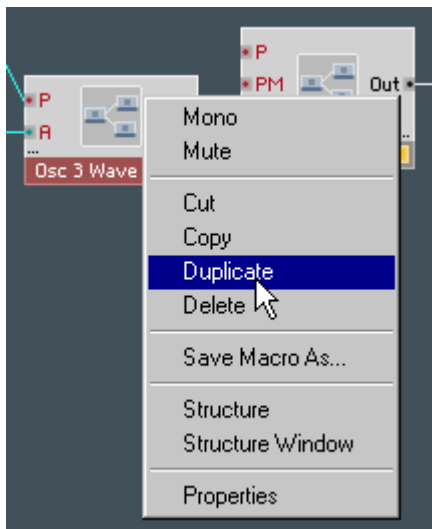
- 2) Drag each macro (by its title at the top of its frame) to its own area in the panel;



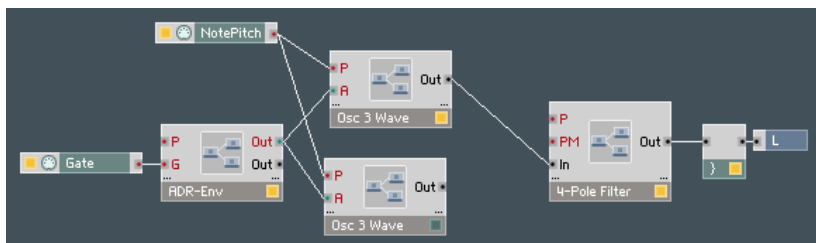
- 3) click the **Lock/Unlock Panel** button again to re-lock the panel. You should now be able to see all three macros clearly: **Osc 3 Wave**, **ADR-Env**, and **4 Pole Filter**.



On considering these components you may wonder if a one-oscillator synth is the be all and end all. After all, it's well known that two oscillators simply give a fatter sound. OK, so let's add another one. In the Instrument structure, XP: Right-click / OS X: Ctrl+click on the macro **Osc 3 Wave** (careful, don't hit one of the ports), select **Duplicate** from the context menu. Done.



It is important to maintain a clean design – especially when dealing with a complex synthesizer. It's easy to create a chaotic layout and wind up spending hours in search of a problem's cause. So - if you didn't do so before like in the screenshots above - let's clean up the Structure window a bit. Move the **4 Pole Filter** macro to the left of the **Out** port, with the **Audio Voice Combiner** ( } ) module in-between. And place the two **Osc 3 Wave** macros neatly above one another to the left of the **4 Pole Filter** macro.



For the next step, let's remove some of the confusion between the two oscillators, which at present are identical, even sharing the same name. Let's give them different labels. To do so, XP: Right-click / OS X: Ctrl+click on the upper **Osc 3 Wave** macro and choose **Properties** from the context menu. A window will appear with a field called **Label** in the top left; here you can



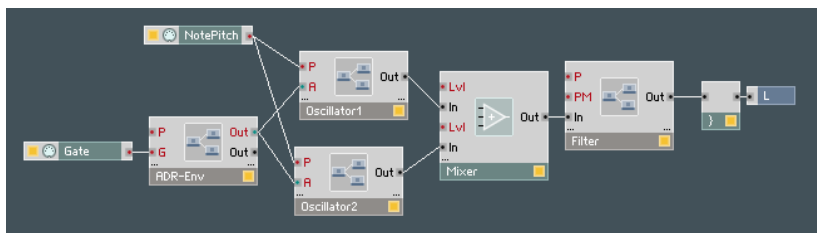
enter a new name, say, **Oscillator1**. Do the same for the lower **Osc 3 Wave** macro, but label it **Oscillator2**. You can also rename the macro **4 Pole Filter** to simply **Filter**.

Now we want the signals of both oscillators to enjoy treatment by the filter, so we need a second wire from **Out** port of **Oscillator2** to the **In** port of **Filter**. We clearly have a problem here, because REAKTOR does not allow you to connect two wires to the same port. Of course that's not really surprising, because you can't put two jack plugs in the same socket either. The solution to this problem is to be found with a little bit of thinking. What we are looking for is a component that can simply combine the signals from the two oscillator macros and pass the sum on to the **In** port of **Filter**. And this component, a mixer for audio signals, is the **Amp/Mixer** module.

To insert the **Amp/Mixer** module, XP: Right-click / OS X: Ctrl+click on a blank part of the Structure window and in the context menu choose **Built-In Module** ⇒ **Signal Path** ⇒ **Amp/Mixer**.



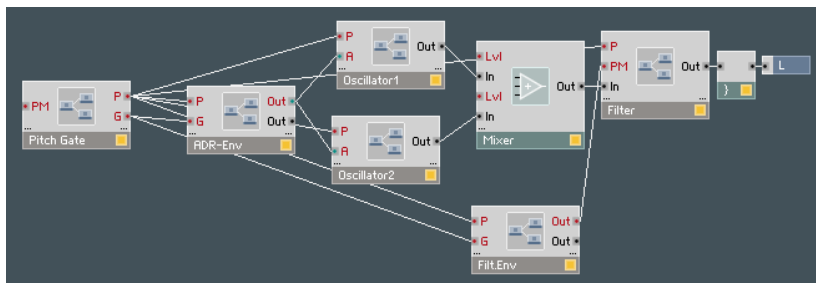
Now place the **Amp/Mixer** module between **Oscillator1 / Oscillator2** and **Filter** (we want things to be neat), and connect the output of **Oscillator1** to the **In** port of **Amp/Mixer**. We need to connect the output of Oscillator2 to **Amp/Mixer** also. But **Amp/Mixer** only has one In port and, as we know, you cannot connect two wires to the same port. Fortunately, this is not a problem, because the **Amp/Mixer** module supports dynamic Input port handling. Simply XP: Ctrl+drag / OS X: X+drag (i.e. hold down Ctrl/APPLE while dragging) from the **Out** port of **Oscillator2** to a spot just below the occupied **In** port of **Amp/Mixer**. This causes **Amp/Mixer** to create another In port, to which you should now complete your connection. The rest is child's play: a wire from the output of **Amp/Mixer** to **In** of the Filter, and a wire from **Out** of the Filter to the input of the **Audio Voice Combiner**. Done.



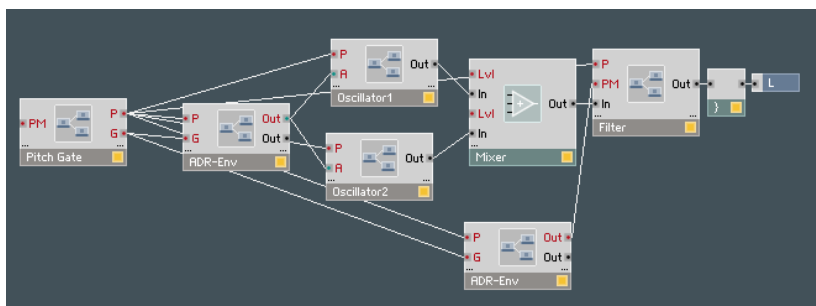
As soon as you complete the last connection, the status LEDs of all modules should light up to indicate that we now have a functional structure.

If you did not use the Duplicate option to get a copy of Oscillator 1 (see above) but did the good old Copy/Paste action you have to connect the out ports of the modules **Gate** and **Note Pitch** to the **P** and **A** in ports of **Oscillator2** as you did for **Oscillator1** to transmit MIDI notes to **Oscillator2** also.

An extra ADR envelope for the Filter makes our synthesizer sound richer. Duplicate the **ADR-Env** macro and assign it to the filter by connecting its upper **Out** (red) port to the **PM** input of the **Filter** macro. Rename it to **Filt.Env**



And, finally, let's exchange the two modules **NotePitch** and **Gate** for a single macro which does the same but also has an integrated module for Pitchbending. Delete **NotePitch** and **Gate** from the structure. Select **Macro** ⇒ **Building Blocks** ⇒ **Pitch+Gate** ⇒ **Pitch + Gate** to insert the **Pitch Gate** macro into the structure. Connect the **P** output of **Pitch Gate** to the **P** inputs of the two **ADR-Env** macros, **Oscillator 1** and **Oscillator 2**, and **Filter**. The **Pitch Gate** **G** output has to be connected to the **G** inputs of both **ADR-Env** macros.



Now the synthesizer can be played properly. The pitch and volume of the incoming MIDI notes are recognized by the synth, and even the pitchbend wheel on the MIDI keyboard works, because the **Pitch Gate** macro is set up to handle all of these tasks.

## Arranging the Panel

Have a look now at the **Instrument** panel. You see a bunch of knobs that are wrapped up within frames to form groups. Each frame corresponds to one of the macros that we have inserted, so we know exactly which controls belong to the **Filter**, which to **Oscillator2**, etc.

Now you can start polishing the panel design. For example, at the moment the controls for **Oscillator2** are still overlaying those for **Oscillator1**. To change this, unlock the panel by clicking the **Lock/Unlock Panel** button (wrench icon) in the instrument panel header. You can tell that the panel is unlocked, because the **Lock/Unlock Panel** button is lit, and the panel is overlaid with a grid. (Note that you can play an instrument when its panel is unlocked, but you cannot change any of its control settings.) Drag all five macros to suitable locations in the panel.



Possible result after polishing the panel layout

Once you are happy with the layout you can freeze it in its current state to make sure that knobs or frames aren't moved inadvertently. Simply click again on the **Lock/Unlock Panel** button to lock the panel.

## Saving

You probably want to save your two-oscillator synthesizer instrument so that you can reuse it in another ensemble. First, let's name it something other than its default name, **Instrument**. Double-click on the name **Instrument** in the panel header to open its Properties dialog. In the Label field, type **My DIY Synth** (or similar) and press the **Enter** key. Now let's save the instrument. XP: Right-click / OS X: Ctrl+click on your new name in the panel header and choose **Save Instrument as...** from the context menu.

In the Save Instrument dialog that appears, choose a folder in which to store your instrument file, specify a file name (or use the one that REAKTOR suggests: **My DIY Synth**) and click on **Save**. When you are asked later whether you want to save the ensemble, you can say **No** because the only part of the ensemble that's worth keeping (the **My DIY Synth** instrument) has already been saved.

---

**Note:** To save the entire ensemble, rather than just one instrument in the ensemble, use **File⇒Save Ensemble...** from the main menu

---

## Luxury

If after some time you feel like adding more features, rest assured that REAKTOR isn't going to limit your urge for experimentation. Just take a look at the macros which are included in the demo. You will find a lot of possibilities to transform this simple synthesizer into a luxurious sound machine. First, try to change the Envelope type from ADR to ADSR to change the percussive sound of your first DIY synth and get something like a lead synthesizer.

## 4.3. Your First DIY Structure

Our DIY synthesizer project was executed mainly using prebuilt macros. We would now like to introduce you to the art of constructing a synthesizer completely from scratch. Contrary to the recommendation we gave above, which was to always separate larger functional units into macros, this new synthesizer will be constructed entirely with modules in a single Structure window, no macros. The main reason for this is the fact that our new device will be of a quite modest nature. It will consist of so few components that any further subdivision into macros would probably cause confusion rather than make things clearer.

### Building the Basic Structure

Select **File⇒New Ensemble** from the main menu to open a new ensemble. In the Ensemble Structure window, delete the default **Instrument**. All that should be left is the default **Master** instrument (containing Level and Tune controls), and the **Audio In** and **Audio Out** modules.

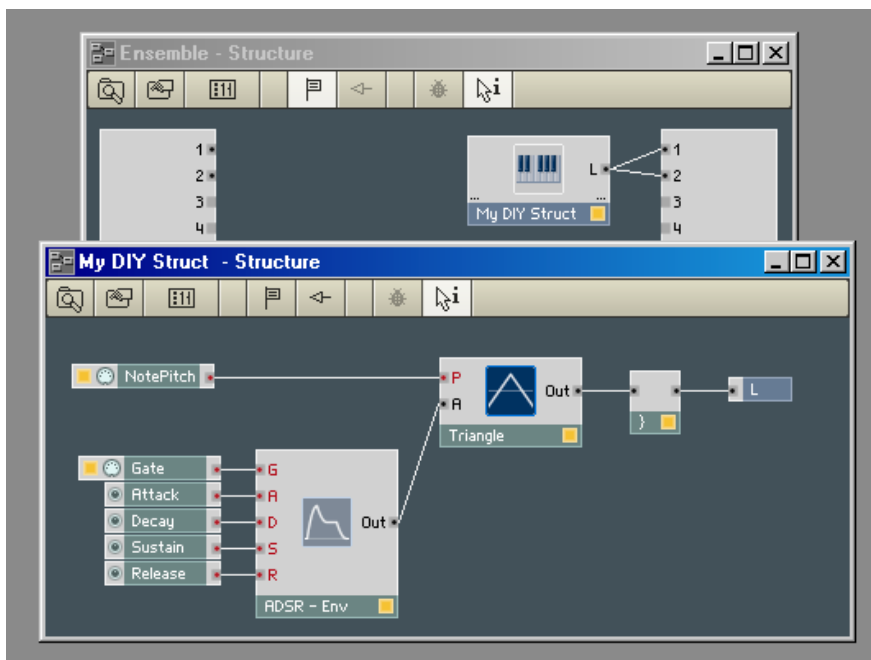
XP: Right-click / OS X: Ctrl+click on a blank part of the Structure window and choose **Insert instrument ⇒ New - 2In2Out** from the context menu. An empty instrument named **Instrument** appears in the structure. Rename **Instrument** to **My DIY Struct** (or similar). Double-click **My DIY Struct** to open its structure and delete all modules inside it except for the **L** output terminal and its connected **Audio Voice Combiner ( }**). Double-click on a blank part of the **My DIY Struct** structure to move one level up to the Ensemble structure. Connect the **L** output port of **My DIY Struct** to input ports **1** and **2** of **Audio Out**.

Open the **My DIY Struct** structure by double-clicking its icon in the Ensemble structure. It is here, in the **My DIY Struct** Structure window that we will implement our synthesizer circuitry.

First we insert an oscillator. Our choice this time is an oscillator module that generates a triangle wave. XP: Right-click / OS X: Ctrl+click on a blank part of the Structure window and in the context menu choose **Built-In Module** ⇒ **Oscillator** ⇒ **Triangle**.

The next step is to add modules that will tell the synthesizer about the volume (gate) and pitch of the incoming MIDI notes. To that end, use the context menu to select **Built-In Module** ⇒ **MIDI In** ⇒ **Gate** and then **Built-In Module** ⇒ **MIDI In** ⇒ **Note Pitch**. For the envelope we choose an **ADSR-Env** module (**Built-In Module** ⇒ **LFO, Envelope** ⇒ **ADSR**).

Now position and interconnect the modules according to the following illustration. Use this shortcut to create the **Attack, Decay, Sustain, Release** controls that connect to the **A, D, S, R** input ports of the **ADSR-Env** module: XP: Right-click / OS X: Ctrl+click on each of the input ports and select **Create Control** from the context menu. Experienced builders use this “trick” all the time to create input-port controls (that they then modify, as necessary, to meet their needs).



## How does it all work?

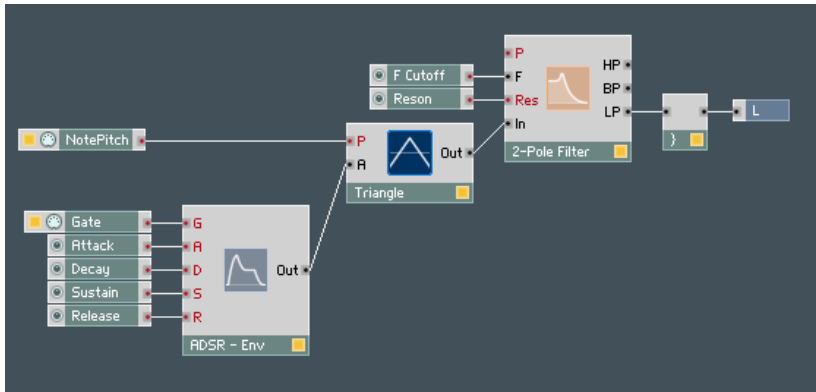
On studying this structure, the following functionality becomes apparent: The **ADSR-Env** module generates an envelope whose shape is specified with the **Attack**, **Decay**, **Sustain** and **Release** knobs in the **My DIY Struct** panel window (keep it tidy!). The envelope is triggered by a rising signal at the gate input **G**, in our case generated by the press of a MIDI key. The pitch of the incoming MIDI note is received by the **NotePitch** module and sent to the **Triangle** module through its **P**(itch) input.

You can already play this synthesizer, but you will very likely soon get tired of the sound - other than the volume envelope there's simply nothing that can be adjusted. The whole thing, however, becomes much more interesting when a filter is brought into play.

## Adding a Resonant Filter

Use the context menu to insert a 2-pole filter (with FM) into **My DIY Struct** structure (**Built-In Module** ⇒ **Filter** ⇒ **Multi 2-PoleFM**). Then connect **Out** of the **Triangle** module to **In** of the **2-Pole Filter** module, and **LP** of **2-Pole Filter** to the **Audio Voice Combiner** ( **}** ).

Next, using **Create Control** (as discussed above), create controls for the **2-Pole Filter** inputs **F**(requency Cutoff) and **Res**(onance) to make the structure look something like the picture below.



## Filter's Function

Play a few notes on your keyboard while at the same time changing the position of the knobs **F Cutoff** and **Reson** in the **My DIY Struct** panel window. (To see these knobs, you'll have to unlock the panel, arrange the controls, and lock it again. If the Lock/Unlock Panel button is not visible in an instrument header, because the header is too narrow, use the header's context menu to select **Lock/Unlock Panel**.)



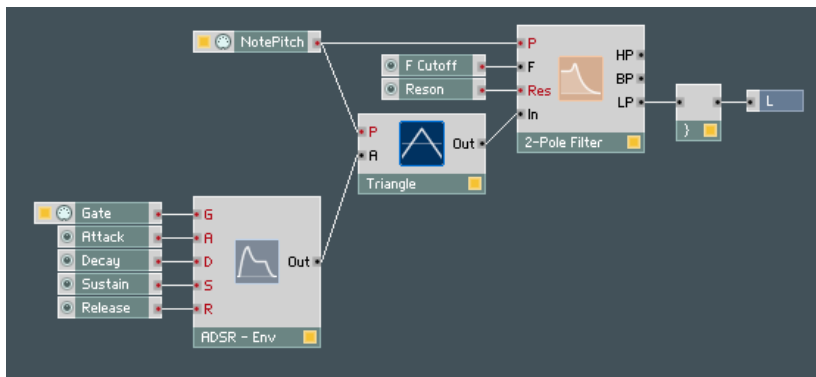
**F Cutoff** sets the filter's cutoff frequency. When utilizing the output **LP** (low pass) of the **2-Pole Filter** module, as done here, all frequencies above the cutoff frequency are removed. By using the other outputs of **2-Pole Filter** it can also be employed as a band pass (**BP**) or high pass (**HP**) filter.

**Reson** sets the resonance of the filter. The higher the resonance value, the more the frequencies near the cutoff frequency are boosted. If you set **Reson** very high, say  $\geq 9.5$ , the filter will begin to self-oscillate. Beware: This can cause extremely loud feedback-type sounds which can, potentially, damage your speakers (and ears)!

## Adding Key Tracking

Now play some low notes and then some high notes on your MIDI instrument and you will notice that the high notes sound relatively dull. This is because the filter operates at a fixed cutoff frequency. This means that no matter what pitch you play, the filter always removes all frequencies above the fixed cutoff frequency. So if you play a note whose frequency is above this cutoff, almost nothing will be heard. We can change this by matching the filter frequency to the respective note pitch. Simply connect the **NotePitch** module with a second wire to the **P**(itch) input of the filter module.

The **2-Pole Filter** circuit is designed to add the control signal at the **P** input to the frequency control signal at the **F** input. The sum of the two then determines the filter's cutoff frequency. If you play some high notes they will sound as you would expect.



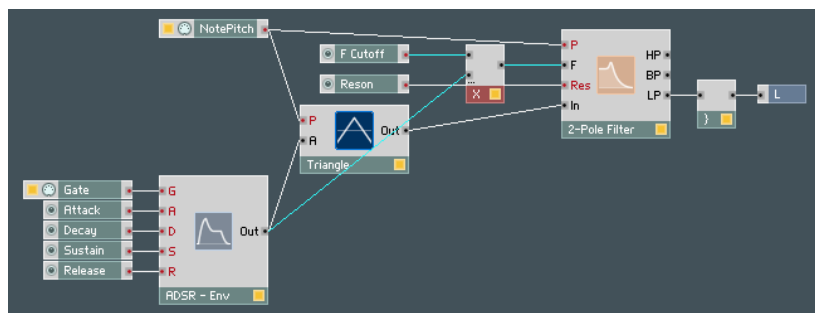
## Adding a Filter Envelope

Finally, we also want to control the filter cutoff frequency with an envelope. For simplicity's sake we will let the existing **ADSR-Env** module take over this task, too. If you wanted to have a more sophisticated synth, you could add a separate envelope just for the filter.

For the envelope to affect the filter cutoff we first need another component, a multiplier (**Built-In Module**  $\Rightarrow$  **Math**  $\Rightarrow$  **Multiply**). Connect **Out** of **ADSR-Env** to



one of the two input ports of **Multiply (X)**, the output of **F Cutoff** to the other **Multiply (X)** input, and the **Multiply (X)** output to the **F** input of **2-Pole Filter**, as illustrated below.



Play a few bars and you will hear that the envelope now affects the filter's cutoff frequency. It works like this: The **ADSR-Env** module outputs a control signal between 0 and 1. This signal is multiplied by the current value of the knob **F Cutoff**. When the envelope is at its maximum value (1), the filter's input **F** receives the value  $1 \times \text{F Cutoff} = \text{F Cutoff}$ . When the envelope reaches its minimum value (0), the signal at **F** is reduced to zero ( $0 \times \text{F Cutoff} = 0$ ).

The function that **F Cutoff** now performs is commonly known as “Envelope Modulation Depth”. To take this into account in the display, open the **F Cutoff** Properties dialog by double-clicking on its module and change the label **F Cutoff** to **Env Mod**.

## Variations

Here are some suggestions for modifications you could make to the structure we have just built:

- Try out the **HP** and **BP** outputs of the **2-Pole Filter** module.
- Replace the **2-Pole Filter** module with a **4-Pole Filter** module.
- Try out different envelopes.
- Add an extra envelope for the filter module.

So you have other ideas? Go ahead and try them. Always remember that in contrast to working with hardware components, there's never any danger of breaking anything in REAKTOR.

Be warned, however, that unexpected - and potentially very loud - sounds will be generated from time to time as you experiment with REAKTOR. To protect both your speakers and your ears, set your initial amplification levels low.

All right! Now get cracking!

## 5. Basic Operation

The REAKTOR user interface follows the conventions of your computer's operating system, so it is easy for someone who has already worked with OS X or Windows to get used to the software. Nevertheless, we want to explain some particular characteristics of REAKTOR and draw your attention to some features that may be new to you.

### 5.1. Mouse

Practically all functions in REAKTOR can be carried out using the mouse. The main operations that will be performed are the following:

- **Selecting** an object is done by clicking on it with the left mouse button. Selected objects (instruments, macros, modules, etc.) are recognized by their title bar which is colored red. If you want to select several objects, hold down the XP: **Ctrl** key or the OS X: **Shift** key on your computer keyboard while clicking on the desired objects one after the other. Alternatively you can click with the left mouse button on a blank part of the window and open a frame by dragging with the button pressed. All objects within the frame are selected.
- **Moving** an object is done by clicking the left mouse button on it and keeping the button pressed while dragging the mouse pointer, and with it the object, to the desired location. To move several objects together, first select all the desired objects, and then move one of the objects as above. All of them will move together, and all the wiring to other objects will remain intact, merely stretching like rubber bands. On releasing the mouse button the modules are aligned on a grid and then remain at the new position. The grid helps to ensure a tidy appearance.
- **Wires** are drawn by clicking and holding the left mouse button on the output port of the object that is to transmit the signal, and then dragging the mouse pointer, and with it the wire, to the desired input port of the object that is to receive the signal. Simply release the left mouse button and the connection is established. Wiring operations can also be carried out in the reverse direction (from input port to output port) the results will still be the same.
- **Double-clicking the left mouse button** on an object (or on the background of a window) will perform various actions, depending on the object. Actions that can be executed by a double-click appear in bold type in an object's corresponding context menu.

- **Clicking the right mouse button (XP) or holding down the Ctrl key and clicking the mouse button (OS X)** opens the context menu that belongs to the object (or window) on which the button was clicked. Context menus play a very important part in REAKTOR's operation, which is why the next section is dedicated to explaining them in detail.

## 5.2. Context Menus

Context menus are lists of commands that are applicable to the object you are clicking on. So, if you want to perform an action on an object or if you need information about it, XP: right-click or OS X: Ctrl+click on it. A context menu will appear whose entries apply to the selected object. Click (with the left mouse button) on a menu item to select it. The menu will disappear and the operation will be carried out. For example, you can delete a module by selecting the entry **Delete** from its context menu.

## 5.3. Key Commands

Many functions in REAKTOR can be performed with keys or key combinations in addition to the mouse. Available key commands are listed in the menus next to the command.

## 5.4 Ensemble Panel and Structure Windows

The REAKTOR workspace comprises two windows: the Ensemble Panel window and the Structure window. The Ensemble Panel window contains the ensemble panel and all of the ensemble's instrument panels. The Structure window contains the structure (internal wiring) of the currently selected object (ensemble, instrument, primary macro, core cell/macro).

There is one Ensemble Panel window. By default, there is one Structure window also. You can, however, choose to open multiple Structure windows by **Alt +** double-clicking the desired objects, or selecting **Structure Window** from the **context menu**. Though we recommend that you work with a single Structure window to keep your screen (and mind!) uncluttered, REAKTOR allows you to open as many separate Structure windows as you like.

Here are some guidelines for managing your REAKTOR Ensemble Panel and Structure windows:

- To open the Ensemble Panel window, choose **View->Show Panel** from the main menu. Or, if you are working in a Structure window, use this trick: Double-click on a blank part of the Structure window to display its

parent Structure window. Keep doing this until you reach the Ensemble Structure window. Double-click there and REAKTOR will display the Ensemble Panel window.

- To open a Structure window, double-click on the desired object (to open it in the shared Structure window) or **Alt +** double-click on the object (to open it in a separate Structure window).
- All open Ensemble Panel and Structure windows are listed at the bottom of the **View** menu. To jump to an open window, select it in the list (or, if the window is visible, simply click anywhere within it).
- To jump one structure up in the hierarchy – i.e. to the structure that *contains* the current structure – double-click on a blank part of the Structure window.
- You move, resize, minimize, and close REAKTOR windows just as you would for any other windows on your platform. If a window is too small to display its entire contents, scrollbars at its right and bottom edges enable you to scroll through the window's contents.

The following applies to the use of REAKTOR in Windows:

- As is the norm with Windows programs, all REAKTOR Ensemble Panel and Structure windows are contained inside the main REAKTOR application window. When this main window is resized, minimized, or covered by another application, all the contained windows are affected.
- When an Ensemble Panel or Structure window is maximized, it expands to fill the entire main REAKTOR window, and all other windows are also maximized until any one of them is reset to a smaller size.
- When a window is minimized, it appears as a small rectangular box at the bottom of the main window.
- To step through all open windows, use **Ctrl + Tab**.

## 6. Menus

In addition to the various **context menus**, the commands for using REAKTOR are accessible from the menu bar of the main window. The program's global functions, controlled from the menu bar, are described below.

### 6.1. File Menu

#### New Ensemble

Selecting **File->New Ensemble** (or pressing XP: **Ctrl + N** / OS X: **X + N**) creates a new ensemble that contains a **Master** instrument and **Audio In** and **Audio Out** modules.

#### Open...

Selecting **File->Open...** (or pressing Windows: **Ctrl + O** / OS X: **X + O**) loads an ensemble file (\*.ens) stored on your disk.

#### Save Ensemble

Selecting **File->Save Ensemble** (or pressing XP: **Ctrl + S** / OS X: **X + S**) stores the current ensemble together with all its instruments, structures, panels, and snapshots in an \*.ens file.

#### Save Ensemble As...

Selecting **File->Save Ensemble As...** (or pressing XP: **Ctrl + Shift + S** / OS X: **X + Shift + S**) is identical to Save Ensemble (see above), but it enables you to specify a new filename and/or folder for the ensemble.

#### Save Window As...

Selecting **File->Save Window As...** (or pressing XP: **Ctrl + E** / OS X: **X + E**) enables you to (re)name and store the contents of the currently selected window.

If the Ensemble Panel window is selected, the ensemble will be saved (in an \*.ens file), just as if you had used the **Save Ensemble** menu command.

If a instrument structure window is selected, the instrument containing the structure will be saved (in an \*.ism file) together with all its structures, panels, and snapshots.

If a Macro structure window is selected, the macro containing the structure will be saved (in an \*.mdl file).

## Import MIDI File...

There is an integrated MIDI File Player in REAKTOR that enables the import and playback of MIDI files in the Standard MIDI File format (SMF). Such MIDI files can be produced by nearly every sequencer program. Under Windows they have the file name extension .mid.

Because it has an integrated MIDI File Player, REAKTOR can play arrangements without a separate sequencer. This option can be especially appealing to live performers: A sequencer running in the background on the same computer could cause glitches and make your performance more difficult, since you would have to load new files into the sequencer as well as into REAKTOR. On top of that, you would then have to alternate between the two programs in order to access important parameters.

There is another advantage to using the integrated MIDI File Player instead of an external sequencer: sample accurate timing. All notes in a MIDI file that begin at the same time will be played by REAKTOR simultaneously, so the timing is perfectly tight. Of course, the MIDI file's timing depends on the accuracy and resolution of the sequencer it was created on.

The REAKTOR MIDI File Player can be loaded either manually or automatically: For manual operation, use **File->Import MIDI File...** from the main menu to load a MIDI file from your disk. For automatic operation, REAKTOR will load a MIDI file upon opening an ensemble if that file is in the same folder and has the same name as the ensemble (but with the extension .mid); for example, mySynth.ens and mySynth.mid.

In the **Settings** menu there are three entries for navigating the MIDI File Player. When **Play MIDI File** is enabled, the MIDI file is played back when you start the REAKTOR clock (by clicking the **Start/Restart Clock** button in the **Ensemble Panel Toolbar**). The MIDI file will be played in an endless loop when **Loop MIDI File** is enabled. You can use this for instance to keep repeating a pattern or sequence of patterns. Finally, **Ignore Tempo Change**, when enabled, causes REAKTOR to ignore all tempo in the MIDI file and play the file back with the tempo set by the REAKTOR clock (BPM).

The transport functions of the MIDI File Player are controlled from the REAKTOR clock:

- Click the **Start/Restart Clock** button to start MIDI File playback from the beginning or to restart playback at the place where the file was paused.
- Click the **Pause/Stop Clock** button once to pause playback of the MIDI File. Click the **Pause/Stop Clock** button a second time to stop playback and rewind the MIDI file back to the beginning.

## Batch Processing

**Batch Processing** enables batch conversion of REAKTOR 3 files to the REAKTOR 5 format, and the analysis of audio files for the granular sampler modules. Simply plug in your REAKTOR 3 USB key, select a source and destination folder, and click on **OK**.

## Recent Ensembles

With a simple mouse click, you can open any one of the eight most recently accessed ensembles.

## Exit

**Exit** closes the REAKTOR program and all its windows, including those in the taskbar. If any changes have been made to the current ensemble since it was last saved, REAKTOR asks if you want to save the file before exiting.

## 6.2. Edit Menu

### Undo

Selecting **Edit->Undo** (or pressing XP: **Ctrl + Z** / OS X: **X + Z**) reverses the effect of the last editing operation carried out in any of the structures. The Undo function does not apply to panel control setting changes; i.e. changing the value of a knob or fader. For this, you want the Compare function in the Snapshots window.

You can set the maximum number of consecutive **Undo** commands in the **Preferences dialog -->Options** page. If your computer runs low on memory, try reducing this number.

### Redo

Selecting **Edit->Redo** (or pressing XP: **Ctrl + Y** / OS X: **X + Y**) reverses the effect of the most recent **Undo** operation (i.e. it “undoes” the last **Undo**). You can execute **Redo** as many times as you previously executed **Undo** until you wind up back where you started.

### Cut

Selecting **Edit->Cut** (or pressing XP: **Ctrl + X** / OS X: **X + X**) cuts (removes) the current selection and copies it to the clipboard. From there it can be inserted in another place using the **Paste** command (**see below**).

## Copy

Selecting **Edit->Copy** (or pressing XP: **Ctrl + C** / OS X: **X + C**) copies the current selection to the clipboard. From there it can be inserted in another place using the **Paste** command (**see below**).

## Paste

Selecting **Edit->Paste** (or pressing XP: **Ctrl + V** / OS X: **X + V**) copies the current contents of the clipboard into the selected structure.

When using the keyboard shortcut for pasting, you can specify where to paste to by clicking in the desired Structure window.

## Duplicate

Selecting **Edit->Duplicate** (or pressing XP: **Ctrl + D** / OS X: **X + D**) creates a copy of the current selection. It is equivalent to selecting Copy, then Paste.

## Delete

Selecting **Edit->Delete** (or pressing the **Del** key) deletes the current selection. You can also use **Delete** in the context menu of the selected object (module, wire, etc.).

## Select All

Selecting **Edit->Select All** (or pressing XP: **Ctrl + A** / OS X: **X + A**) selects all the objects in the current window. You can then unselect individual objects by XP: Ctrl+clicking / OS X: X+clicking on them.

# 6.3. Settings Menu

## Sample Rate

**Sample Rate** sets the sample rate at which REAKTOR generates and processes audio signals. With higher sample rates you can achieve better sound quality, but the CPU load rises proportionally. You can change the internal sample rate to any of the values in the menu. The range of available values depends on your sound card or host plug-in. If the internal sample rate is different from the sound card's or host plug-in's sample rate, the **Audio In** and **Audio Out** modules will do the necessary sample-rate conversion.



## Control Rate

**Control Rate** sets the control rate for REAKTOR event signals; i.e. the number of times per second that event-signal values are updated. The control rate is applied globally to all primary modules that generate or process events; e.g. **LFO**, **Slow Random**, **Event Hold**, **A-to-E**, **Event Smoother**, and more. Since the control rate is very low compared to the sample rate, these modules need very little CPU power. For this reason, good builders choose to work with event signals rather than audio signals whenever possible (i.e. whenever it doesn't degrade the sound).

Higher control rates give a better resolution in time, resulting in finer steps in the signal.

## MIDI Learn

Selecting **Settings->MIDI Learn** (or pressing XP: **Ctrl + T** / OS X: **X+ T**) activates MIDI Learn mode for the currently selected panel control. This mode is automatically deactivated after a MIDI-controller message is received. There is a corresponding **MIDI Learn** button (midi connector icon) in the **Ensemble Panel Toolbar**.

## Set Protected/Set Unprotected

Enables/disables Protection mode. In protection mode only a very limited edit of the ensemble panel and structure is possible. Insertion, deletion, movement, of panel controls and the alteration of voices is disabled.

## Automatic Layout

Enables Automatic Layout mode for all instrument panels. (This is equivalent to turning on **Automatic Panel Layout** in an ensemble's Properties dialog, Appearance page.) By default this option is switched on.

## External Sync

Toggles between the internal REAKTOR clock and an external clock (received via MIDI) for all **Sync Clock** and **1/96 Clock** modules. Also enables control of **Start/Stop** modules by external MIDI-Start/Stop messages. When **External Sync** is turned on, the tempo cannot be adjusted by the master clock BPM field in the Main toolbar; instead, the internal clock is adjusted according to the external clock.

## MIDI Clock Out

If you enable this option, REAKTOR sends out MIDI clock ticks on all active MIDI out ports (as set in the Audio Setup dialog, MIDI page).

## Clock Start

Starts the REAKTOR master clock which controls all of the clock-driven modules in the ensemble. This works with both internal and external clocks. It sets the output of all **Start/Stop** modules to “start”. In the Ensemble Panel toolbar there is a **Start/Restart Clock** button for the same function.

## Clock Stop

Stops the REAKTOR master clock which controls all the **Sync Clock** and **1/96 Clock** modules in the ensemble. This works with both internal and external clocks. It sets the output of all **Start/Stop** modules to “on”. In the Ensemble Panel toolbar there is a **Pause/Stop Clock** button for the same function.

## Play MIDI File, Loop MIDI File, Ignore Tempo Change

These commands act on REAKTOR’s integrated MIDI File Player. For more information about this, please see the Imp.

# 6.4. System Menu

## Run/Stop Audio

With this menu command all audio computations can be started (**Run Audio**) or stopped (**Stop Audio**). In effect, this is the main on/off switch for the REAKTOR software. The same function can be accessed by the **Run/Stop Audio** button in the Main toolbar.

## Debug

The **Debug** menu provides four options: **Measure CPU Usage**, **Show Module Sorting**, **Show Event Initialization Order**, and **Optimization**.

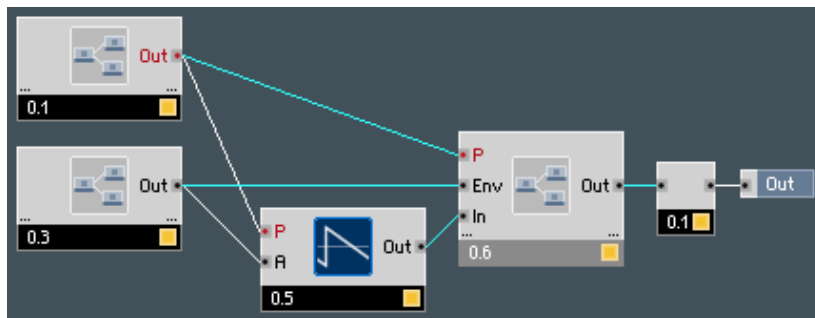
## Measure CPU Usage

The **Measure CPU Usage** option switches all audio-processing components (instruments, macros, modules) to CPU-load measuring mode. The current CPU load is displayed in black labels on the components (in structure view). This feature is useful for determining how much of the total load is being caused by each component. This information can help you to streamline the structure, thus allowing for the generation of more voices.

Some components do not have a number displayed on them, i.e. they keep their normal label. That's because they do not actually use up any CPU power for audio processing, either because they are not active or because they only do event processing.

The displayed value may differ a little from the actual CPU load in normal operation.

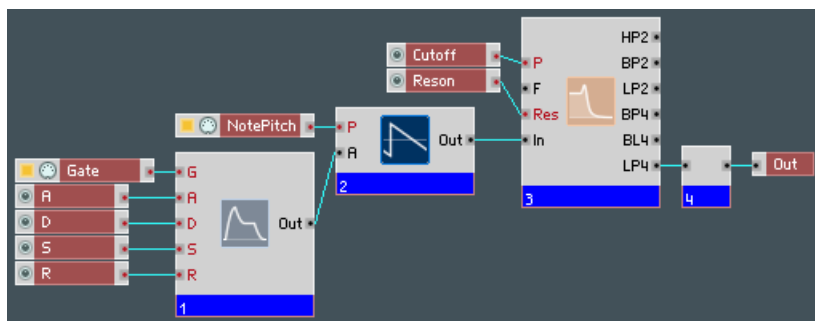
This mode is only available when **Run Audio** is active. During load measuring the audio output is switched off. You can enable the **Measure CPU Usage** option with the key combination XP: **Ctrl + U** / OS X: **X + U**.



CPU Usage Display

## Show Module Sorting

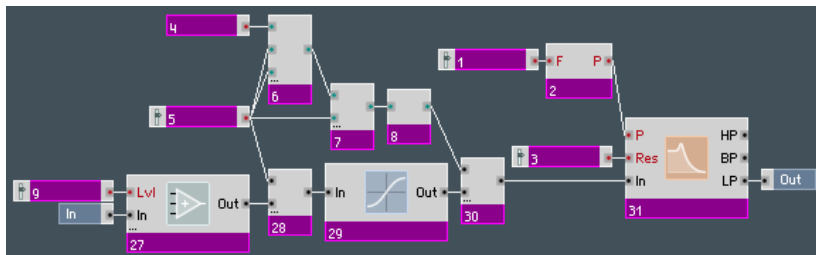
The **Show Module Sorting** option switches all audio-processing modules to sorting mode. In this mode, the current position of each module within the overall stream of audio processing is shown. This position will be displayed as a number in blue label on each module.



Show module Sorting mode

## Show Event Initialization Order

This **Show Event Initialization Order** option displays the current position of each module within the overall stream of event initialization.



Show Event Initialization Order screenshot

## Audio + MIDI Settings...

This menu item opens a dialog for selecting your audio and MIDI interfaces. Detailed instructions are found in the chapter **REAKTOR Standalone Version**

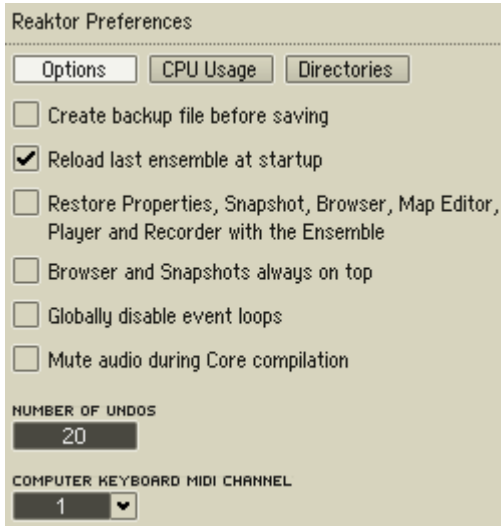
## OSC Settings...

This menu item opens the OSC Setup dialog. Detailed instructions on how to use OSC are found in **Open Sound Control** chapter.

## Preferences

This menu item opens the Preferences dialog in which you will find a number of REAKTOR preference options that you can set as desired. The Preferences dialog consists of three pages: Options, CPU Usage, and Directories.

## Options



- If you enable **Create backup file before saving**, existing files will not be overwritten on **Save** or **Save As...** but will be kept with the filename extension **.bak**. In this way, you can always return to the previously saved version of a file if something should go wrong (by renaming the extension **.bak** to the original extension).
- **Reload last ensemble at startup** tells REAKTOR to load the ensemble that was active when it was last shut down.
- When enabled, **Restore Properties, Snapshots, Browser, Map Editor, Player and Recorder with the Ensemble** restores an ensemble's Properties dialog and Snapshots window, Browser, Sample Map Editor, Playerbox and Recorderbox just as they were when the ensemble was last saved.
- **Browser and Snapshots always on top** causes the Browser and Snapshots window to display on top of all other REAKTOR dialogs and windows.
- When turned on, **Globally disable event loops** prevents event-signal loops from occurring in ensembles. (If an event loop is about to occur, REAKTOR displays a message revealing the source of the loop and asking you how to proceed.) Event loops can lead to stack overflow crashes, which in turn can make ensembles un-playable and, in some cases, un-openable. If this happens, restart REAKTOR, turn on **Globally disable event loops**, open the problematic ensemble, and trace the source of the event loop with the help of event-loop identification

messages. (It can be useful to disable audio to prevent further loops occurring during this process.) We recommend that you globally disable event loops to maximize the stability of REAKTOR. To ensure backward compatibility, files saved in older versions of REAKTOR have event loops enabled by default.

---

**Note:** In most cases, the Iteration module can avoid the need for creating event loops. The Iteration module has a limited speed option in its properties, which can avoid audio glitches caused by processing a large number of iterations too fast.

---

- **Mute audio during Core compilation**, when enabled, turns off all audio processing during the period of time that it takes to compile a core object. This makes compilation go faster.
- The **Number of Undos** box lets you set the maximum number of undos that can be performed in a row. Setting **Number of Undos** to 20, for example, enables you to undo the last 20 edits you've made by issuing 20 Undo commands in a row. Enter 0 to disable the undo function.

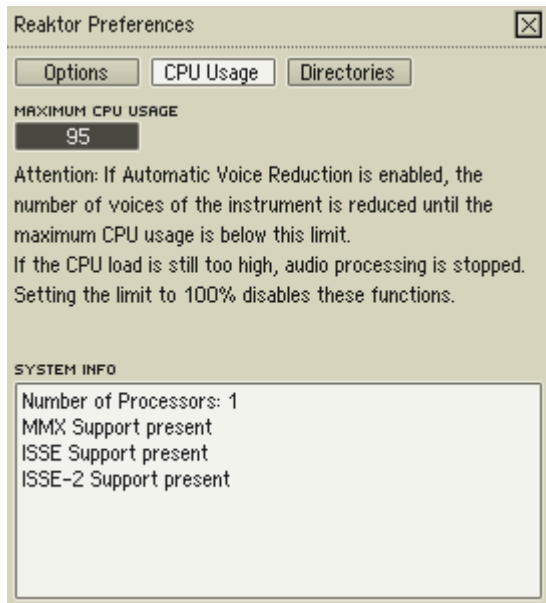
---

**Note:** If you work on ensembles containing large audio files, you will need a lot of computer memory to continue to use the undo function. If you do run out of memory, try reducing the Number of Undos.

---

- **Computer Keyboard MIDI Channel** specifies the MIDI channel for MIDI notes played at the computer keyboard. This is set to 1 by default, because REAKTOR instruments are set to receive MIDI messages on channel 1 by default. If, however, you want to use the keyboard to play an instrument that receives MIDI on a different channel (e.g. 2), you would set Computer Keyboard MIDI Channel to this channel (2).

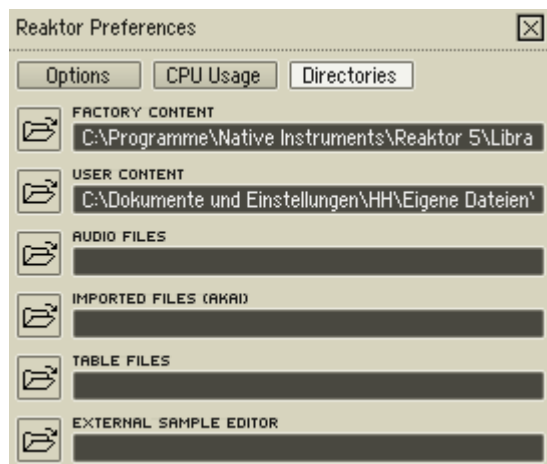
## CPU Usage



REAKTOR can automatically reduce the number of voices in polyphonic instruments when the total CPU load of the ensemble reaches a certain limit. In this way, the polyphony can be adjusted according to the available processing power.

The upper CPU load limit is set by **Maximum CPU Usage** (where **80** = 80%, **95** = 95%, etc.). Note that REAKTOR changes the number of voices only for instruments in which **Automatic Voice Reduction** has been selected (Properties dialog, Function page). To disable automatic voice reduction for all instruments in an ensemble, set **Maximum CPU Usage** to 100.

## Directories



### Factory Content

This path specifies the default folder in which the REAKTOR system ensembles, instruments, primary macros, core cells, and core macros are stored; i.e. the objects that were automatically copied to your hard disk when REAKTOR was installed on your system.

The System Content path is used by all relevant context menus (e.g. Ensemble Structure and Panel windows, instrument and macro Structure windows, etc.) and by the top row of buttons in the Browser (Ens., Instr., Macro, Core. C., Core M.) to enable you to access system objects quickly and easily.

### User Content

This path specifies the default folder in which custom user ensembles, instruments, primary macros, core cells, core macros, audio files, imported files, pictures files, snapshot files, and table files are stored. These are files that you create, acquire, modify, etc.

Just as with the Factory Content path, the User Content path is used by all relevant context menus (e.g. Ensemble Structure and panel windows, instrument and macro Structure windows, etc.) and by the User row of buttons in the Browser (Ens., Instr., Macro, Core. C., Core M.) to enable you to access your custom objects quickly and easily.

---

**Note:** Never store user files in REAKTOR 5 system folders, because REAKTOR updates might delete them!

---



## Audio Files

This path specifies the initial default folder for loading and saving audio files (\*.wav, \*.aif, \*.aiff) from anywhere in REAKTOR: Player, Recorder, Sample Map Editor, Tapedeck and Sampler modules, etc.

## Imported Files (Akai)

This path specifies the default folder for storing Map files (.map) that have been converted with the Akai-Import function.

## Table Files

This path specifies the default folder for loading and saving table files (in the Properties dialog of a Table module). Table files have the file extension \*.ntf and can be used in the Audio and Event Table modules.

## External Sample Editor

Enter the pathname of your favorite sample editor here, and you can launch it by selecting **Edit** from the drop-down Edit Sample List menu in the Sample Map Editor window.

# 6.5. View Menu

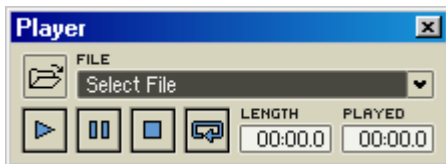
## Show/Hide Hints

Shows and hides the popup hints that appear when the mouse points to an object (toolbar button, module, port, etc.). The keyboard shortcut is XP: **Ctrl + I** / OS X: **X + I**.

## Show/Hide Toolbox

Shows and hides the Main toolbar. For details, see Toolbar. The keyboard shortcut is XP: **Ctrl + F1** / OS X: **X + F1**.

## Show/Hide Playerbox



Shows and hides the Playerbox. The keyboard shortcut is XP: **Ctrl + F2** / OS X: **X + F2**.




The Playerbox enables you to route the audio output from an audio file (\*.wav, \*.aif, \*.aiff) into your REAKTOR ensembles for processing (filtering, delay, reverb, etc.). The Playerbox output audio signal is made available to an ensemble from the top two ports of the Audio In module (in the Ensemble Structure window).

---

**Note:** When you play an audio file in the Playerbox, all non-Playerbox audio signals at the top two ports of the Audio In module will be muted.

---






The Playerbox contains the following controls:

-  **Load:** Loads an audio file from the hard disk.
-  **File** drop-down menu: After having loaded an audio file, the folder from which it came will then be scanned for additional audio files. These files will appear in the **File** drop-down menu for quick access.
-  **Play:** Starts playback of a loaded audio file.

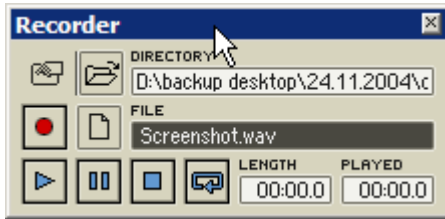
---

**Note:** The audio file will play at the current REAKTOR sample rate, as set in Sample Rate in the Main toolbar.

---

-  **Pause:** Pauses playback of an audio file. Press the Pause button again to proceed with playback.
-  **Stop:** Stops playback and goes back to the beginning of the audio file.
-  **Loop:** Plays an audio file in an endless loop.
-  **Length:** Shows the length of the loaded audio file.
-  **Played:** Shows the current playback position.

## Show/Hide Recorderbox



Shows and hides the Recorderbox. The keyboard shortcut is XP: **Ctrl + F3** / OS X: **X + F3**.

The Recorderbox enables you to record the audio output of an ensemble by writing it to an audio file on your hard disk (XP-systems record .wav files, OS X-systems record .aif files). The Recorderbox records the audio signal that is output to the top two ports of the Audio Out module (in the Ensemble Structure window).


The Recorder can also play an audio file by sending its output to the top two ports of the Audio Out module.

---

**Note:** When you play an audio file in the Recorderbox, all non-Recorderbox audio signals at the top two ports of the Audio Out module will be muted.

---

The Recorderbox has its own Recorder Settings dialog where you can define conditions for starting/stopping the Recorder.


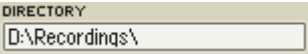








To open the Recorder Settings dialog, click the **Recorder Settings** button  in the upper-left corner of the Recorderbox.



The Recorder Settings dialog provides the following options:

- **Record Start By (Manual, Note On, Clock Start):** When **Manual** is enabled, you start the Recorder by arming it using the Record button hitting the Play or Pause button. When **Note On** is enabled, the Recorder is started by a MIDI Note On event. When **Clock Start** is enabled, the Recorder starts when the REAKTOR clock is started with the Start/Restart Clock button in the Main toolbar.
- **Record Stop By (Manual Only, Note Off, Clock Stop, Loop Length):** When **Manual Only** is enabled, you stop the Recorder using the Stop button. It is also possible to stop the Recorder for a while and resume with the Pause button. When **Note Off** is enabled, the Recorder is stopped by a MIDI Note Off event. When **Clock Stop** is enabled, the Recorder stops when the REAKTOR clock is stopped with the Stop/Pause Clock button in the Main toolbar. When **Loop Length** is enabled, the Recorder is stopped when the loop length duration (as specified by the Loop Length option, see below) has been reached.
- **Start Offset (Bars)** specifies a time delay (in bars) before the start of the recording. This can be useful if you want to record an echo effect in an advanced phase for instance, but you don't want the beginning.
- **Loop Length (Bars)** specifies a duration (in bars) for the recording, when **Loop Length** is selected in the **Record Stop By** section above.
- **Slave Player Controls to Recorder:** When on, this causes the Playerbox transport controls to be slaved to the Recorderbox transport controls.

The Recorderbox contains the following controls:

-  **Load:** Creates a new audio file (or selects an existing one) to record to. Be careful: If you select an existing audio file, the recording will overwrite (delete) the file's previous contents. You can also use **Load** to load an audio file for playback (as described above).
-  **Directory:** Shows the current folder being used for recording.
-  **Record:** Arms the Recorder. Once armed, the actual recording will start when you click the **Pause** button (to un-pause the Recorder).
-  **New:** Creates a new, empty audio file on your hard disk in which to store the recording.
- **File:** Shows the name of the audio file currently being recorded (or about to be recorded). You can change the name of a new or existing audio file here.
-  **Play:** Starts playback of the currently loaded audio file (as displayed in the File field).
-  **Pause:** Pauses/un-pauses recording or playback.
-  **Stop:** Stops recording or playback and resets the recording/playback pointer to the beginning of the file.
-  **Loop:** Plays an audio file in an endless loop.
-  **Length:** Shows the length of the loaded audio file.
-  **Played:** Shows the current record/playback position.

## Show/Hide Properties

Shows and hides the Properties dialog for the selected object (ensemble, instrument, macro, etc.). You can also open an object's Properties dialog by double-clicking on its title bar, or XP: right-clicking / OS X: **Ctrl**+clicking in the title bar and choosing **Object Properties** from the context menu, where **Object** is the name of the clicked-on object: **EchoBox Properties**, **"Fader" Properties**, etc. The Properties dialog always refers to the currently selected object and can stay open while you work. The keyboard shortcut is **F4**.

## Show/Hide Sample Map Editor

Shows and hides the Sample Map Editor window. The keyboard shortcut is **F7**.

## Show/Hide Browser

Shows and hides the Browser. The keyboard shortcut is **F5**.

## Show/Hide Snapshots

Shows and hides the Snapshots window. The keyboard shortcut is **F6**.

## Reset All Tool Window Positions

Resets all the tool windows (Playerbox, Recorderbox, Properties dialog, Sample Map Editor, and Browser) to their default sizes and locations. If you ever have trouble finding one of these windows, selecting **Reset All Tool Window Positions** should fix things right up.

## Show/Hide Panel

Shows and hides the Ensemble Panel window.


## Store Panelset

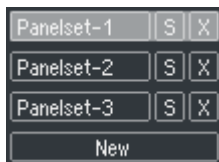
Enables you to store up to 8 different panelsets with an ensemble.

To store a panelset:

1. Create the panelset by arranging your Ensemble Panel window as desired (instrument positions, views, visibilities).

Select **View->Store Panelset->N**, where **N** is a number between 1 and 8. The keyboard shortcut is XP: **Ctrl + Alt + N** / OS X: **X + Alt + N**.

You can also use the  Panelset bar, which is located in the panel Toolbar, to store panelsets. Using the Panelset bar, rather than **View->Store Panelset**, enables you to store as many panelsets with an ensemble as your computer memory will allow.




The panelset bar with three different panelset views stored

## Recall Panelset

Enables you to recall the first 8 panelsets in an ensemble.

To recall a panelset:

1. Select **View->Recall Panelset->N**, where **N** is a number between 1 and 8. The keyboard shortcut is XP: **Crt + N** / OS X: **X + N**.

You can also use the  Panelset bar to recall panelsets. Using the Panelset bar, rather than **View->Store Panelset**, enables you to recall as many panelsets as the ensemble contains (not just the first 8).

## Close All Structures

Closes all open Structure windows. If you work with multiple Structure windows, this is a great way to un-clutter your view.

## Cascade

Cascades all open REAKTOR windows. This function is available under Windows only. On OSX the functions are minimize and close.

## Tile Horizontally

Tiles all open REAKTOR windows horizontally. This function is available under Windows only.

## Tile Vertically

Tiles all open REAKTOR windows vertically. This function is available under Windows only.

## Minimize

This function is available under Mac OS X only.

## Close

This function is available under Mac OS X only.

## Arrange Icons

The function arranges the minimized windows. This function is available under Windows only.

## List of Open Windows

All currently open REAKTOR windows are listed at the bottom of the View menu. Clicking a window in the list selects it, brings it to the front of the REAKTOR workspace, and restores it (if it was minimized).

## 6.6. ? Menu

### About

Choosing **About** from the XP: ? / OS X: **REAKTOR 5** menu opens the About REAKTOR 5 window. On the About page of this window is the REAKTOR version number and your personal REAKTOR serial number. On the other pages are web links for the REAKTOR user library and user forum, program updates, FAQs, and technical support.





## 7. REAKTOR Toolbars

There are three REAKTOR toolbars:

- **Main toolbar** - appears at the top of the REAKTOR application window. It contains tools for managing the REAKTOR program.



- **Ensemble Panel toolbar** - appears at the top of the Ensemble Panel window. It contains tools for working in ensembles.



- **Structure toolbar** - appears at the top of a Structure window. It contains tools for working in structures.



---

**Note:** The command bar at the top of an instrument panel is called a **header**, not a toolbar.






---


### 7.1. Main Toolbar





The Main toolbar


The Main toolbar has the following elements (left to right):


- Clicking on the  **NI icon** or the  **REAKTOR icon** opens the About window.
-  **Open** opens an existing ensemble. You can also press XP: **Ctrl + O** / OS X: **X + O** to do this.
-  **Save** stores the current ensemble with all changes made to it. You can also press XP: **Ctrl + S** / OS X: **X + S** to do this.
-  **Undo** undoes the last editing operation. You can also press XP: **Ctrl + Z** / OS X: **X + Z** to do this.


-  **Run/Stop Audio** turns all REAKTOR audio processing on and off. When you are building or editing an ensemble and don't need sound, you can turn off audio processing to minimize your CPU load.

-  **CPU Load Display** displays the percentage of your CPU's processing capacity that is being used to run REAKTOR. In instances of CPU overload, the display will show **Over**. (The maximum allowed CPU usage is set in the CPU Usage page of the **Preferences** dialog.) Remember, a computer running REAKTOR has to do more than process audio! It also has to transfer audio to the sound card, process MIDI data, execute events, and draw REAKTOR's graphical displays, as well as run the computer operating system and any other programs that might be open. This is why REAKTOR can reach its limits at **CPU Load** levels well below 100%. The threshold for smooth ensemble operation is normally between 60% and 80%. To test your computer's limits, raise the number of voices in an ensemble until the CPU overload message appears.

- The  **Sample Rate** list box specifies the sample rate at which REAKTOR is running. You can select a different sample rate from the list; only those rates supported by your sound card are shown.

- The  **Audio In** level meter displays the level of the audio that is being input into REAKTOR; i.e. the audio signal (from the Playerbox or your sound card) that is made available to ensembles at the top two ports of the Audio In module.










- The  **Audio Out** level meter displays the level of the audio that is being output from REAKTOR; i.e. the audio signal generated by an ensemble that is made available to the outside world (sound card, speakers, headphones, host program for REAKTOR running as a plug-in, etc.) at the top two ports of the Audio Out module.


- The  **MIDI In** lamp lights whenever REAKTOR receives a MIDI event from an active MIDI Input port (as set in the MIDI page of the Audio Setup dialog).
- The **MIDI Out** lamp lights whenever REAKTOR sends a MIDI event to an active MIDI Output port (as set in the MIDI page of the Audio Setup dialog).

## 7.2. Ensemble Panel Toolbar



The Ensemble Panel toolbar








- The  **Show/Hide Panelset Bar** button shows and hides the Panelset bar.
- The  **Snapshots** button opens the Snapshots window.
- The  **Browser** button opens the Browser.
- The  **Properties** button opens the Properties dialog.
- The  **Ensemble Structure** button opens the Ensemble Structure window.
- The  **Pause/Stop Clock** button stops the REAKTOR master clock. If a MIDI file has been imported (**File->Import MIDI File**); pushing **Pause/Stop Clock** once pauses the MIDI file playback; pushing **Pause/Stop Clock** a second time stops the playback and rewinds the MIDI file back to its beginning.
- The  **Start/Restart Clock** button starts the REAKTOR master clock. If a MIDI file has been imported, **Start** starts MIDI File playback from the beginning or restarts playback at the place where the file was paused.
- Use the  **Tempo** selector to adjust the rate of the REAKTOR master clock in beats per minute (BPM). Double-click on the tempo display to enable direct entry of numerical values from your computer keyboard. Use the up/down arrows to adjust the tempo in 1 BPM increments.
- Use the  **MIDI Learn** button to quickly and easily assign a panel control (knob, fader, etc.) to an external MIDI controller (keyboard mod wheel, knob, fader, etc.). To do so: Select a panel control by clicking on it; click on the **MIDI Learn** button; operate the external MIDI controller (e.g. move the keyboard mod wheel). To cancel the controller assignment, open the control's Properties dialog and deselect its **Activate MIDI In** option.

- The  **Show/Hide Info** button shows and hides popup information. When **Show Info** is on (button lit), pointing your mouse to an object (instrument, primary or core macro/module, etc.) shows information on the object in a popup, and pointing to a wire shows the current values being sent along the wire. When **Hide Info** is on (button unlit), pointing to objects and wires shows no popups.

## 7.3. Structure Toolbar

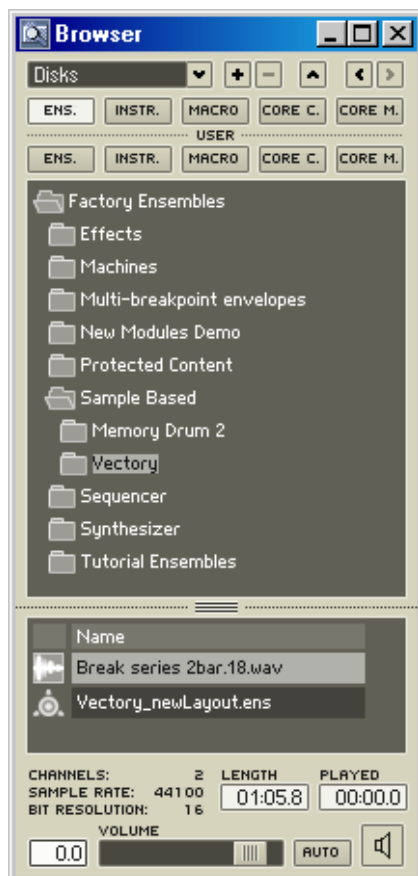


The Structure toolbar

- The  **Browser** button opens the Browser.
- The  **Properties** button opens the Properties dialog.
- The  **Ensemble Panel** button opens the Ensemble Panel window.
- The  **Bookmark** button bookmarks the current Structure window so that you can jump straight to it from any other Structure window in the ensemble.
- The  **Jump to Bookmark** button jumps from any Structure window in the ensemble to the bookmarked Structure window (if the **Jump** arrow is pointing toward the **Bookmark** button), or from the bookmarked Structure window to the last-displayed Structure window (if the **Jump** arrow is pointing away from the **Bookmark** button).
- The  **Debug** button is active within REAKTOR Core Cells only.
- The  **Show/Hide Info** button shows and hides popup information. When **Show Info** is on (button lit), pointing your mouse to an object (instrument, primary or core macro/module, etc.) shows information on the object in a popup, and pointing to a wire shows the current values being sent along the wire. When **Hide Info** is on (button unlit), pointing to objects and wires shows no popups.

## 8. The Browser

The Browser enables you to quickly and easily access the following types of files for use in REAKTOR:



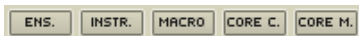
The Browser

- Audio files (\*.wav, \*.aif, \*.aiff), ensemble files (\*.ens), instrument files (\*.ism), primary macro files (\*.mdl), core cell files (\*.rcc), and core macro files (\*.rcm) by using the **Disk Navigation** controls.



The **+** button adds a folder to the favorites list. The favorites list is appended to the **Disks** list.

- Factory (REAKTOR) ensembles, instruments, macros, core cells, and core macros by using the top row of **Ens.**, **Instr.**, **Macro**, **Core C.**, and **Core M.** buttons.



- Custom (user) ensembles, instruments, macros, core cells, and core macros by using the bottom (User) row of **Ens.**, **Instr.**, **Macro**, **Core C.**, and **Core M.** buttons.



## 8.1. Accessing Files

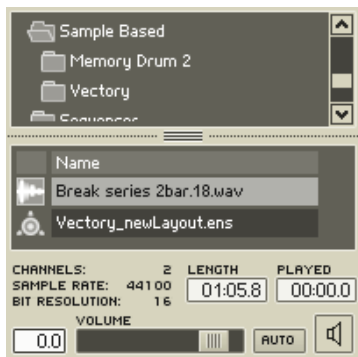
To open the Browser: select **View->Show Browser** from the main menu; or click on the **Show/Hide Browser** button in the Ensemble or Structure toolbar; or press **F5**.

The Browser is organized into two panes:



The upper pane of the browser containing the Disk Navigation










- In the upper pane, you locate files by using the self-explanatory **Disk Navigation** controls, or the **Ens.**, **Instr.**, **Macro**, **Core C.**, and **Core M.** buttons underneath. The top row of **Ens.**, **Instr.**, **Macro**, etc. buttons navigates to the system files that were installed along with the REAKTOR program. The bottom row of buttons navigates to your custom, user files. (The paths REAKTOR needs to find these files are in the **Preferences** dialog, Directories page. If necessary, you can change them there.)




The lower pane of the browser

- The lower pane enables you to access (audition, open, load, drag into structures) files.

You can use the Browser to access the following files:

-  **Audio file:** You can load an audio file (\*.wav, \*.aif, \*.aiff) by dragging it from the Browser (lower pane) to the Sample Map Editor, or to a sampler or tape deck display in an instrument panel. To locate an audio file, use the **Disk Navigation** controls at the top of the Browser.
-  **Ensemble file:** You can open a new ensemble file (\*.ens) by dragging it from the Browser (lower pane) to the REAKTOR workspace. To locate an ensemble file, use either of the **Ens.** buttons.
-  **Instrument file:** You can insert an instrument file (\*.ism) by dragging it from the Browser to the ensemble panel or ensemble Structure window. To locate an instrument file, use either of the **Instr.** buttons.
-  **Primary macro file:** You can load a primary macro file (\*.mdl) by dragging it from the Browser to a primary Structure window. To locate a primary macro file, use either of the **Macro** buttons.
-  **Core cell file:** You can load a core cell file (\*.rcc) by dragging it from the Browser to a primary Structure window. To locate a core cell file, use either of the **Core C.** buttons.
-  **Core macro file:** You can load a core macro file (\*.rcm) by dragging it from the Browser to a core Structure window. To locate a core macro file, use either of the **Core M.** buttons.
-  **Sample map file:** You can load a sample map file (\*.map) by dragging it from the Browser to the Sample Map Editor or to a sampler module panel display. To locate a sample map file, use the **Disk Navigation** controls.
-  **MIDI file:** You can load a standard MIDI file (\*.mid) by dragging it from the Browser to the REAKTOR workspace. (This is equivalent to using **File->Import MIDI File** to load the file.) This loads the MIDI file into REAKTOR's MIDI File Player (for playback using the Pause/Stop and Start/Restart buttons in the Ensemble Panel toolbar). To locate a MIDI file, use the **Disk Navigation** controls.
-  **Table file:** You can load a table file (\*.ntf) by dragging it from the Browser to a table display in an instrument panel. To locate a sample map file, use the **Disk Navigation** controls.

-  **Snapshot file:** You can load a snapshot file (\*.ssf) by dragging it from the Browser to the Snapshots window. To locate a snapshot file, use the **Disk Navigation** controls.

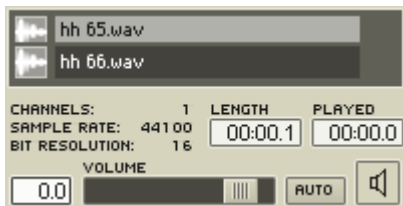
---

**Note:** When you use the Browser (or a context menu) to insert an object (instrument, primary macro, core cell, or core macro) into a structure, the object's input and output ports are not automatically connected to anything (an instrument, macro, Audio In port, Audio Out port, etc.). You must wire all desired object connections manually.


---

## 8.2. Auditioning Files


The Browser supports audio-file auditioning (pre-listening):



The audition section of the browser

1. In the Browser (lower pane, see above section **Browser**), select an audio file (\*.wav, \*.aif, \*.aiff) to audition. Playback controls appear at the bottom of the Browser, along with properties of the selected audio file (channels, sample rate, length, etc.).
2. Click on the  **Play** button (speaker icon) to start/stop auditioning the audio file. Use the **Volume** fader to set the volume.



3. If you turn  **Auto** on, auditioning starts automatically when you select an audio file.



## 9. Ensemble

The ensemble is the highest object in the REAKTOR structural hierarchy. The entire contents of the current REAKTOR workspace (instruments, control settings, audio input/output connections, snapshots, etc.) are stored with the ensemble (in an \*.ens file), and are restored when the ensemble is reloaded.

REAKTOR's structural hierarchy is as follows:

- An ensemble can contain instruments.
- An instrument can contain other instruments, as well as primary macros, primary modules, and core cells.
- A primary macro can contain other primary macros, as well as primary modules, and core cells.
- A core cell can contain core macros and core modules.
- A core macro can contain other core macros, as well as core modules.

In terms of panel display:



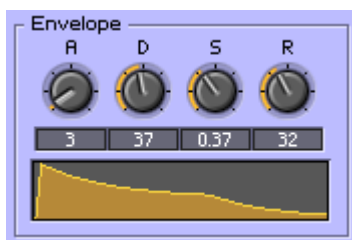
Ensemble Panel Window

- An ensemble has one panel window,



Instrument Panel

- Each instrument has a panel that you can choose to show or hide in the Ensemble Panel window.

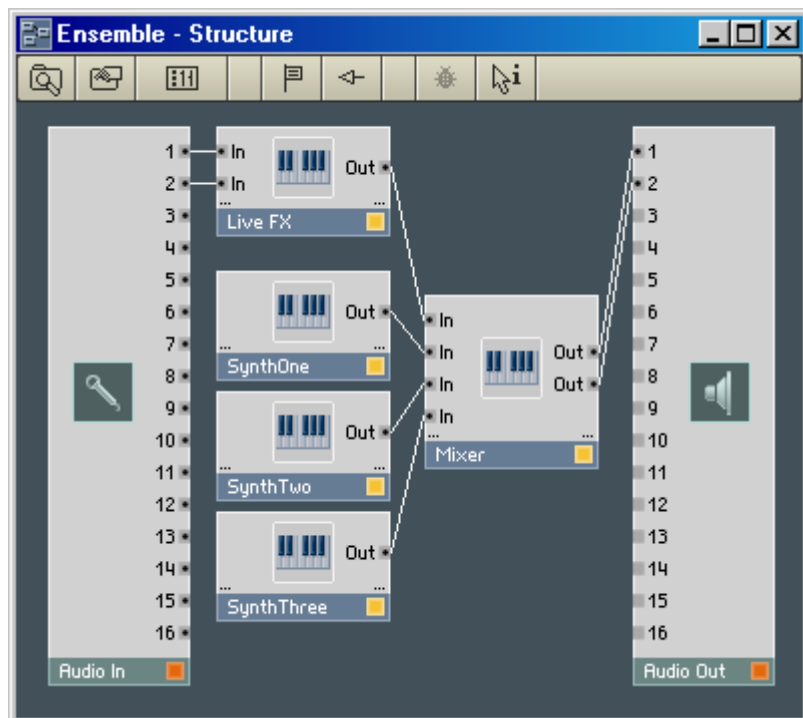


Frame of a macro in the Instrument panel

- Each primary macro can have a frame (that you choose to show or hide) in its instrument panel.

While the above is a simplified explanation, it should give you a clear idea of REAKTOR's basic architecture.

## 9.1. Ensemble Structure Window



Ensemble with five instruments: one live effect, three sound sources and a mixer. At left is the Audio In module.

The Ensemble Structure window gives a bird's eye view of the structure of the entire ensemble. It contains icons of all the instruments in the ensemble, and the Audio In and Audio Out modules, which provide access to your sound card or plug-in host.

### Audio In Module

The **Audio In** module represents your REAKTOR audio inputs, as defined on the **SoundCard** and **Routing** pages of the **Audio Setup** dialog (**System->Audio + MIDI Settings...**). The **Audio In** module is a fixed part of the Ensemble Structure window and cannot be removed.

The context menu of the **Audio In** module contains two entries:

- **Mute** mutes (disables) the **Audio In** module. If you are not routing any audio input into an ensemble, it is good programming (though not required) to mute **Audio In**.
- **Properties** opens the **Audio Setup** dialog (just like **System->Audio + MIDI Settings...**).

The **Audio In** module provides a total of 16 ports for incoming audio (from the Playerbox, an external microphone, etc.). The actual number of available ports (i.e. those to which you can connect wires within an ensemble) is determined by the number of audio inputs your sound card supports. Available ports are marked with a black dot; unavailable ports are gray.

## Audio Out Module

The **Audio Out** module represents your REAKTOR audio outputs, as defined on the **SoundCard** and **Routing** pages of the **Audio Setup** dialog (**System->Audio + MIDI Settings...**). The **Audio Out** module is a fixed part of the Ensemble Structure window and cannot be removed.

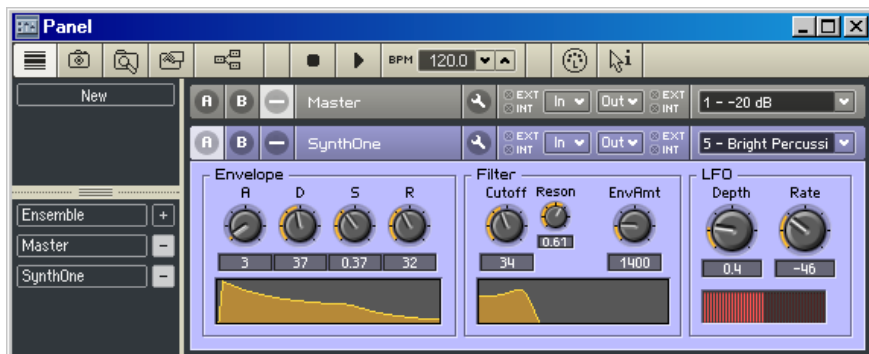
The context menu of the **Audio Out** module contains two entries:

- **Mute** mutes (disables) the **Audio Out** module. If your ensemble does not generate sound (e.g. if it only generates a display), it is good programming (though not required) to mute **Audio Out**.
- **Properties** opens the **Audio Setup** dialog (just like **System->Audio + MIDI Settings...**).

The **Audio Out** module provides a total of 16 ports for outgoing audio to your sound card of plug-in host. The actual number of available ports (i.e. those to which you can connect wires within an ensemble) is determined by the number of audio outputs your sound card supports. Available ports are marked with a black dot; unavailable ports are gray.

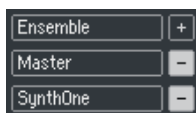
## 9.2. Ensemble Panel Window


The Ensemble Panel window can display all (or one, or none, etc.) of the instrument panels in the ensemble.



Ensemble Panel Window with its Panelsetbar on the left side

To show/hide instrument panels, you use the **Panelset bar**.



To show/hide the **Panelset bar**, click the  **Show/Hide Panelset bar** button in the Ensemble Panel toolbar.


Each instrument panel has three views: A, B, and minimized. You select these views by clicking the **A**, **B**, and **Minimize** (—) buttons in the instrument header.



The A and B views are set in the **Properties** dialog **Appearance** pages of the instrument controls (knobs, faders, XYs, macro frames, etc.). The minimized view displays the instrument header alone.

## 9.3. Ensemble Properties Dialog

There are many ways to open an ensemble's Properties dialog. Use whichever one suits you best:

- Windows XP: Right-click / OS X: Ctrl+click on a blank part of the Ensemble Panel window and select **Ensemble Properties** from the context menu.
- Click on the  **Show/Hide Properties** button in the Ensemble Structure window toolbar.
- Click on a blank part of the Ensemble Panel window and click on the  **Show/Hide Properties** button in its toolbar.
- Click on a blank part of the Ensemble Panel window and select **View->Show Properties** (or press **F4**).

Like all Properties dialogs, the Ensemble Properties dialog has four pages, each accessible from a picture button on the top of the dialog:



**Function,**



**Info,**



**Appearance,** and



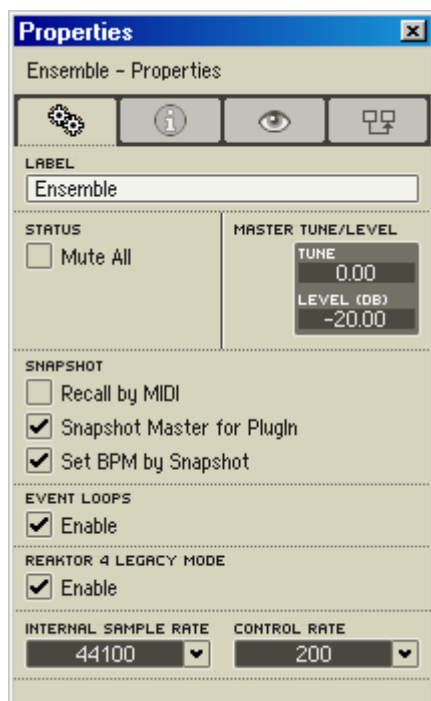
**Connection.**

---

**Note:** The contents of the Properties dialog automatically changes to display the values for the currently selected object (ensemble, instrument, primary macro, etc.). So, to compare object values, leave the Properties dialog open and toggle your selection between the two objects.

---

## 9.3.1. Function Page



Ensemble Properties dialog, Function page

### Label

The **Label** field contains the label “Ensemble” and cannot be changed. Note that the **Label** field of all other REAKTOR objects (except for core modules) can be changed, enabling you to customize names of instruments, primary macros, core cells, etc.

### Status

**Mute All** mutes (disables) all instruments in the ensemble. This reduces REAKTOR’s CPU usage to a minimal level (just enough to keep REAKTOR running), marks all instrument icons (in the Ensemble Structure window) with a red M (for Mute), and draws a red X over all primary output/input ports.

## Master Tune/Level

**Tune** adjusts the ensemble's global pitch. The value is specified in fractional semitone units (12 semitones = an octave). Positive values raise the pitch; negative values lower it.

**Level** changes the ensemble's global volume level. The value is specified in dB (6 dB doubles/halves the volume). Positive values raise the level, negative values lower it.

## Snapshot

When **Recall by MIDI** is enabled, an incoming MIDI Program Change message with the value N (where N is an integer from 0-127) will recall the snapshot with the value N+1 (if that snapshot exists). Thus a Program Change message of 0 will recall snapshot 1, a message of 1 will recall snapshot 2, and so on. This way you can quickly and easily recall snapshots from your MIDI controller (keyboard) by issuing MIDI Program Change messages of the desired snapshot number.

When **Snapshot Master for Plug-In** is enabled the ensemble snapshots are available in host programs. There can only be one snapshot master. This setting can alternatively be activated for single instruments.

**Set BPM by Snapshot** enables REAKTOR master clock BPM settings to be saved/recalled with snapshots.

## Event Loops

When the **Event Loops** option is enabled, REAKTOR allows event-signal loops to occur within the ensemble. These loops can lead to stack overflow crashes, which in turn can make the ensemble un-playable and, in some cases, un-openable.

When **Event Loops** is disabled, event-signal loops are prevented from occurring. If a loop is about to occur, REAKTOR displays a message revealing the source of the loop and asking you how to proceed.

We recommend that you disable **Event Loops** to maximize the stability of REAKTOR. To ensure backward compatibility, ensemble files saved in older versions of REAKTOR have **Event Loops** enabled by default.



## REAKTOR 4 Legacy Mode

REAKTOR 5 has a new initialization scheme for event inputs that is used if the **REAKTOR 4 Legacy Mode** option is disabled. We strongly recommend that you disable **REAKTOR 4 Legacy Mode** in your ensembles for the sake of future compatibility!

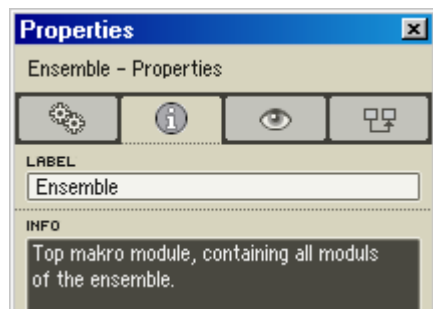
## Internal Sample Rate

**Internal Sample Rate** sets the sample rate at which REAKTOR generates and processes audio signals. With higher sample rates you can achieve better sound quality, but the CPU load rises proportionally. You can change the internal sample rate to any of the values in the menu. The range of available values depends on your sound card or host plug-in. If the internal sample rate is different from the sound card's or host plug-in's sample rate, the **Audio In** and **Audio Out** modules will do the necessary sample-rate conversion.

## Control Rate


**Control Rate** sets the control rate for REAKTOR event signals; i.e. the number of times per second that event-signal values are updated. The control rate is applied globally to all primary modules that generate or process events; e.g. **LFO**, **Slow Random**, **Event Hold**, **A-to-E**, **Event Smoother**, and more. Since the control rate is very low compared to the sample rate, these modules need very little CPU power. For this reason, good builders choose to work with event signals rather than audio signals whenever possible (i.e. whenever it doesn't degrade the sound).

## 9.3.2. Info Page

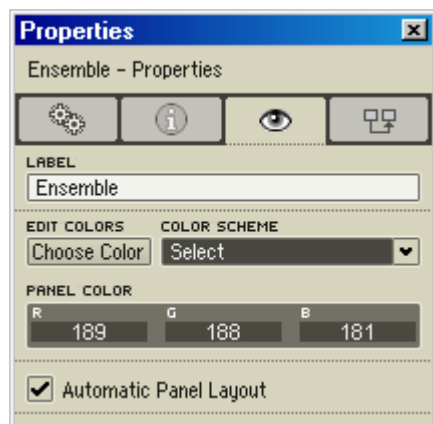


Ensemble Properties dialog, Info page

Enter desired information about your ensemble into the **Info** field of this page.

If  **Show Info** in the Ensemble Panel or Structure toolbar is enabled, your **Info** field text will be displayed in a popup whenever the mouse points to the ensemble panel header.

## 9.3.3. Appearance Page



Ensemble Properties dialog, Appearance page

### Edit Colors

- **Choose Color:** Click this button to use the Color dialog palette to choose a color for the ensemble panel header. Note that an ensemble panel consists only of a header; there is no ensemble panel “body.”

- **Color Scheme: Set to Custom** sets the ensemble's color scheme to the current custom color scheme. **Save as Custom** saves the ensemble's color scheme as the new custom scheme. Note that REAKTOR only supports one custom color scheme; if you save a new scheme, you'll overwrite the previous scheme. **Set to Default** sets the ensemble's color scheme to the default scheme (grey panel with orange indicators).
- **Panel Color:** You can use the Color dialog palette to choose an ensemble panel header color (see **Choose Color**, above). Or you can mix your own color by using the **R**, **G**, and **B** fields to enter values for the color's red, green, and blue components. Each field accepts values from 0 (none) to 256 (full). Entering 0 for all three fields creates black; entering 256 creates white. And so on.

### **Automatic Panel Layout**

The **Automatic Panel Layout** option, when enabled (its default setting), causes all panels displayed in the Ensemble Panel window to be arranged tidily within the window. When **Automatic Panel Layout** is disabled, the panels can be placed anywhere within the window.

## 9.3.4. Connection Page



Ensemble Properties dialog, Connection page

### MIDI

- The **MIDI In Device** drop-down menu specifies which of the available MIDI In devices the ensemble can receive messages from. (You make MIDI In devices available to ensembles in the Audio Setup dialog, MIDI page.) Typically, **MIDI In Device** is set to **All**, enabling the ensemble to receive messages from all available MIDI In devices. In some cases, however, you might want to prevent an ensemble from receiving messages from a certain MIDI In device.
- **Channel** specifies the MIDI Channel number used by the ensemble for MIDI input. The ensemble receives only those MIDI messages that are sent on the specified MIDI Channel number.
- **Morph** activates the ensemble snapshot morph.
- **Controller** specifies the controller number used for the snapshot morph.

## OSC

- The **OSC Source** drop-down menu lets you choose the OSC source(s) from which the ensemble receives OSC data.
- The **OSC Target** drop-down menu lets you choose the OSC target(s) to which the ensemble sends OSC data.
- **OSC Connections** lists the active OSC connections.

## External Sync

When **External Sync** is enabled: an external clock signal (received via MIDI) controls all **Sync Clock** and **1/96 Clock** modules in the ensemble; MIDI Start/Stop messages control all **Start/Stop** modules; and the clock tempo cannot be adjusted in the BPM field on the Main toolbar.

---

**Note:** You can also enable/disable **External Sync** from the REAKTOR Settings menu.

---

When **External Sync** is disabled: the internal REAKTOR master clock controls the **Sync Clock** and **1/96 Clock** modules; the REAKTOR **Pause/Stop Clock** and **Start/Restart Clock** buttons control the **Start/Stop** modules; and the clock tempo can be adjusted in the BPM field.

# 10. Instruments

A REAKTOR instrument is an object that has its own internal structure, MIDI processing, control panel, and snapshots. In the Ensemble Structure window, instrument objects can be recognized by their blue label and keyboard icon.



Instrument object

An instrument can contain other instruments, as well as primary macros, primary modules, and core cells. It may also be called a different name.

## 10.1. Adding Instruments to an Ensemble

You add instruments to an ensemble by loading them from the REAKTOR system library or from your user content storage area (set in Preferences dialog, Directories page).

To add an instrument to an ensemble you can use any of these methods:

- XP: Right-click / OS X: Ctrl+click on a blank part of the structure to which you want to add the instrument. Normally, this is the ensemble structure; but you can also add instruments to instrument structures.
- Use **Insert Instrument** from the context menu to find and insert the desired instrument.
- Open the Browser (**View->Show Browser** or **F5**). Use the upper **Instr.** button to find a system instrument, or the lower **Instr.** button to find a user instrument (from your user content storage area). In the lower pane, drag the desired instrument to the structure.
- Use the Browser's **Disk Navigation** controls (top row) to navigate to the desired instrument folder, then drag the instrument to the structure.

The system library provides a generous selection of premade sound-generation and effects instruments. If you want to start developing a new instrument, you first need to load an empty one (i.e. one that begins with **\_New**) from the system library.

When inserting an instrument in a structure, you are in effect creating a copy of the instrument file that is stored on your disk. The instrument copy and the instrument file are completely independent. Changes you make to the instrument copy will not affect the instrument file, and vice-versa.



If you want to change the instrument file, you must change the copy (in an ensemble), and then use **Save Instrument As...** (from the instrument's context menu) to save the changed copy over the existing instrument file.

---

**Note:** Instruments can also be inserted within another instrument.

---

## 10.2. Ports

There is no fixed arrangement of input and output ports for instruments. The type and number of ports in an instrument is determined by the user by the insertion of **Terminals** (  ,  ) in the instrument structure.

The connections into and out of an instrument through the terminals must always be **monophonic** (rather than polyphonic). For this reason, if the instrument is polyphonic, an **Audio Voice Combiner** module needs to be inserted before the output port(s) to convert the polyphonic signal to a monophonic signal before it is output.

## 10.3. Context Menu

The context menu of an instrument object in the Ensemble Structure window contains the following entries:

- **Mute**, when on, disables the selected instrument.
- **Solo**, when enabled, connects the output of the selected instrument directly to the Audio Out module (i.e. to the sound card or plug-in host). All instruments that lie upstream from the selected instrument (i.e. that feed into the selected instrument) remain active. All instruments that lie downstream from the selected instrument (i.e. into which the selected instrument feeds) are muted. For example, say a Synth instrument signal is fed to a Chorus instrument, the Synth + Chorus signal is then fed to a Compressor instrument, and the Synth + Chorus + Compressor signal is finally connected to the Audio Out module. If the Chorus instrument is in **Solo** mode, the Synth + Chorus signal is connected to the Audio Out module. The Compressor is not included in the circuit, because it lies downstream from the **Solo** Chorus instrument (i.e. Chorus feeds into Compressor).
- **Cut** removes the selected instrument from the structure and stores it temporarily in the clipboard. From there, the instrument can be pasted (using the **Paste** command) to a different structure (or another location in the same structure) .





- **Copy** does the same thing as **Cut**, but it does not remove the instrument from the structure.
- **Duplicate** creates a copy of the selected instrument in the same structure. Choosing **Duplicate** is equivalent to choosing **Copy**, then **Paste**.
- **Delete** deletes the selected instrument from the structure.
- **Save Instrument As...** enables the selected instrument to be saved to an \*.ism file on your disk. Builders typically use **Save Instrument As...** to save new or modified instruments to their user content Instruments folder.
- **Structure** opens the selected instrument's structure in the main structure window. Choosing **Structure** is equivalent to double-clicking on the instrument's structure icon.
- **Structure Window** opens the selected instrument's structure in a separate (i.e. not the main) structure window. Doing so enables multiple structure windows to be open at the same time. Choosing **Structure Window** is equivalent to Alt+double-clicking on the instrument's structure icon.
- **Properties** opens the selected instrument's Properties dialog. For details, see Instrument Properties below.

## 10.4. Instrument Header


The Instrument header contains the following elements:







The Instrument header

- Instruments have two panels views: A and B. The  **A** and  **B** buttons in the instrument header enable you to choose which of these panel views to display. You can specify if an object (knob, fader, meter, etc.) is to appear in panel A, B, both, or neither by using the **A**, **B**, **AB**, and **Visible** options in its **Properties** dialog (Appearance page).
- Clicking on the  **Minimize** (—) button hides the instrument panel and leaves only the header visible. Clicking on **Minimize** (—) again redisplay the panel.
- The instrument name displays the text in the  **Label** field of the instrument's Properties dialog.



- The  **Lock/Unlock Panel** button locks and unlocks the instrument panel. Locking a panel freezes all its elements (controls, displays, etc.) in their current locations. Unlocking a panel enables you to move elements to new locations. Note that, when locked, panel controls can be adjusted (e.g. knobs can be turned), but not moved (to different panel locations); when unlocked, panel controls can be moved, but not adjusted.



- The four MIDI activity lamps – **External** and **Internal** MIDI In, and **External** and **Internal** MIDI Out – light when external/internal MIDI events arrive at the active MIDI In ports or are sent to the active MIDI Out ports. (You configure your REAKTOR MIDI ports in the **Audio Setup** dialog, MIDI page.)
- The  **In** and **Out** drop-down menus enable you to set all the instrument's input and output connections (MIDI and wiring).
- The  **Snapshot** drop-down menu enables you to recall snapshots for the instrument.
-  **Voices** displays (and allows you to change) the number of polyphonic voices allocated to the instrument. This value can also be changed in the **Voices** field of the instrument's **Properties** dialog (Function page).
-  **Unison** displays (and allows you to change) the maximum number of unison voices per note that the instrument will play. The richness and chorus-like quality of the “unison” effect (made famous by hardware synthesizers) is enabled by setting **Unison** to 2 or higher. The unison voices are detuned (with respect to one another) by the value specified by **Unison Sprd** in the instrument's **Properties** dialog (Function page). The minimum number of unison voices per note can be set there as well (**Min Unison V.**).

---

**Note:** REAKTOR enables you to save the current **Unison** value with a snapshot, i.e. to specify a different number of maximum unison voices for each snapshot. But the current **Voices** value applies to the entire instrument (all snapshots).

---

## 10.5. Instrument Properties

There are many ways to open an instrument's Properties dialog. Use whichever one suits you best:

- Double-click on the instrument's title bar (not its keyboard icon!) in any structure window. Or double-click on the instrument's name in its panel header.
- XP: Right-click / OS X: Ctrl+click on the instrument in any structure window, and select **Properties** from the context menu. Or XP: Right-click / OS X: Ctrl+click on the instrument's name in its panel header, and select **Properties** from the context menu
- Select the instrument in the Ensemble Panel or any structure window, and select **View->Show Properties** from the main menu (or press **F4**).

Like all Properties dialogs, an instrument's Properties dialog has four pages, each accessible from a picture button on the top of the dialog:



**Function,**



**Info,**



**Appearance, and**



**Connection.**

---

**Note:** The contents of the Properties dialog automatically changes to display the values for the currently selected object (ensemble, instrument, primary macro, etc.). So, to compare object values, leave the Properties dialog open and toggle your selection between the two objects.

---

## 10.5.1. Function Page

**Properties**

**Instrument - Properties**

**LABEL**

Instrument

**STATUS**

☐ Solo

☐ Mute

**TUNING**

TUNE  
0.00

UNISON SPRO  
0.05

**VOICE ALLOCATION**

VOICE & MIDI SLAVE TO  
None

☐ Lock Voices

VOICES	MAX UNISON V	MIN UNISON V
1	1	1

☐ Automatic Voice Reduction

**VOICE ASSIGN**

☐ Oldest ☐ Reassign

☐ Newest

☐ Nearest

**SNAPSHOT**

☐ Store by Parent ☐ Only if changed

☐ Recall by Parent ☒ Recall by MIDI

☐ Snapshot Master for Plugin

**EVENT LOOPS**

☐ Enable

Properties dialog of an instrument, Function page

### Label

The **Label** field specifies the name of the instrument; i.e. the name that appears in the instrument panel header. You can change this field to (re)name your instrument.

## Status

- **Solo**, when enabled, connects the output of the instrument directly to the Audio Out module (i.e. to the sound card or plug-in host). All objects that lie upstream from the instrument (i.e. that feed into the instrument) remain active. All objects that lie downstream from the instrument (i.e. into which the instrument feeds) are muted. For an example, see above, **Context Menu**.
- **Mute** mutes (disables) the instrument and all objects that lie upstream from it (i.e. that feed into it). In structure view, an instrument whose Mute option is turned on has a red M over its status LED and red crosses over its input/output ports.

## Tuning

- **Tune** adjusts the instrument's pitch with respect to the master ensemble tuning (as set in the **Ensemble Properties** dialog, Function page). The value is specified in fractional semitone units (12 semitones = an octave). Positive values raise the pitch; negative values lower it. A typical application is to detune two instruments to get a fatter sound; a good value for this is **Tune** = 0.05 (equivalent to 5 cents or 1/20th of a semitone).
- **Unison Spread** determines the degree of detuning between each of the instrument's unison voices. (Note that the instrument must have 2+ unison voices for **Unison Spread** to have an effect.) As with **Tune**, the **Unison Spread** value is specified in fractional semitone units. A typical value is 0.05, which detunes each of the unison voices by 5 cents (1/20th of a semitone) with respect to one another, making for a fatter sound.

## Voice Allocation

Each instrument has its own polyphonic voice allocation settings.

- **Voice & MIDI Slave To**, when turned on, enables the instrument's voice allocation and MIDI In settings to be controlled from another instrument in the ensemble.
- **Lock Voices**, when enabled, locks the instrument's voice allocation settings (**Voices**, **Max Unison V**, and **Min Unison V**). If you need to change any of these settings, simply disable **Lock Voices**.
- **Voices** specifies the total number of polyphonic voices that the instrument can play. This number applies to all the polyphonic modules in the instrument (i.e. all modules whose Properties dialog Mono option is disabled).

- **Max Unison V.** specifies the maximum number of unison voices that an instrument can play per note. If, for example, you set **Max Unison V.** to 3, the instrument will play a maximum of 3 unison voices per note. (The minimum number is set by **Min Unison V.**, see below.) The amount of detuning between voices is set with **Unison Spread** (see above).
- **Min Unison V.** specifies the minimum number of unison voices that an instrument can play per note. If, for example, you set **Min Unison V.** to 2, the instrument will play a minimum of 2 unison voices per note.

The **Voices**, **Max Unison V.**, and **Min Unison V.** values are interdependent. **Voices** specifies the total number of polyphonic voices that the instrument can play. If the instrument does not make use of unison playing, **Max Unison V.** and **Min Unison V.** are both set to 1. If the instrument does make use of unison, **Max Unison V.** and **Min Unison V.** specify the maximum and minimum number of unison voices that the instrument can play per note.

**For example, say the Voices, Max Unison V., and Min Unison V. settings were 24/1/1;** the instrument could play up to 24 notes at the same time, with no unison effect. **Say the settings were 24/3/3;** the instrument could play up to 8 notes at the same time with 3 unison voices per note ( $8 * 3 = 24$ ). **Now say the settings were 24/3/2;** the instrument could play up to 8 notes at the same time with 3 unison voices per note ( $8 * 3 = 24$ ), or up to 12 notes at the same time with 2 unison voices per note ( $12 * 2 = 24$ ). And so on.

If **Max Unison V.** and **Min Unison V.** are different, (e.g. 3 and 2, as in our example), REAKTOR automatically switches between the **Max** and **Min** values depending on how many notes are being played at the same time. With a 24/3/2 configuration, if the number of notes being played at the same time is  $\leq 8$ , each note will have 3 unison voices; if the number of notes being played at the same time is  $> 8$ , each note will have 2 unison voices.

- When **Automatic Voice Reduction** is enabled, REAKTOR will automatically reduce the number of instrument voices (specified by **Voices**) when the CPU load exceeds the limit set in the Preferences dialog (CPU Usage page). This way, polyphony can be adjusted according to the available processing power.

## Voice Assign

When the number of voices in an instrument is not sufficient to process all the notes being played at the same time, REAKTOR needs to choose intelligently which voice (or voices, if the instrument is in **Unison** mode) to “steal” from an existing note and reassign to a new note. There are three options for such voice assignment: Oldest, Newest, and Nearest.

- When **Oldest** is enabled, the voice which has been held longest is stopped and assigned to the new note. This is the most common voice assignment strategy.
- When **Newest** is enabled, the most recently played voice is stopped and assigned to the new note. This can be useful for playing a melody over held notes, because none of the held voices will get stopped and reassigned.
- When **Nearest** is enabled, the voice closest in pitch to the new note is stopped and assigned to the new note. This is good when using polyphonic portamento (glide).
- **Reassign** determines what happens when the same note is played again. Either the voice already playing the note is reused or another voice is used. **Reassign** mode is good for making efficient use of a limited number of voices, and it is also what you are used to from playing the piano.

## Snapshot

You find more about the principle Snapshots in REAKTOR in the **Snapshot** section.

- When **Recall by Parent** is enabled and **Store by Parent** is disabled, you can recall an instrument's snapshots by recalling a snapshot in its parent object (usually the ensemble, but sometimes another instrument). For example, let's say an instrument's **Recall by Parent** is enabled and **Store by Parent** is disabled. If you store the snapshot **iSnap1** in the instrument, then store **eSnap1** in the ensemble, recalling **eSnap1** in the ensemble automatically recalls **iSnap1** in the instrument.
- When **Recall by Parent** and **Store by Parent** are both enabled, you can store and recall an instrument's snapshots by storing and recalling a snapshot in its parent object. For example, let's say an instrument's **Recall by Parent** and **Store by Parent** are enabled. If you create the settings for a new snapshot in the instrument, then store the snapshot **Snap1** in the ensemble, a snapshot with the same name (**Snap1**) is stored in the instrument. Recalling **Snap1** in the ensemble automatically recalls **Snap1** in the instrument.

- When **Recall by Parent** is disabled and **Store by Parent** is enabled, you can store an instrument's snapshots by storing a snapshot in its parent object. For example, let's say an instrument's **Recall by Parent** is disabled and **Store by Parent** is enabled. If you create the settings for a new snapshot in the instrument, then store the snapshot **Snap1** in the ensemble, a snapshot with the same name (**Snap1**) is stored in the instrument. However, since **Recall by Parent** is disabled, recalling **Snap1** in the ensemble does not recall **Snap1** in the instrument.
- When **Only if changed** and **Store by Parent** are both enabled, if you store a new snapshot in the parent (ensemble), a snapshot with the same name will only be stored in the child (instrument) if the settings for the new snapshot in the child are different from the settings for the current snapshot. This saves space in the instrument's snapshot list.
- When **Recall by MIDI** is enabled, an incoming MIDI Program Change message with the value N (where N is an integer from 0-127) will recall the snapshot with the value N+1 (if that snapshot exists). Thus a Program Change message of 0 will recall snapshot 1, a message of 1 will recall snapshot 2, and so on. This way you can quickly and easily recall snapshots from your MIDI controller (keyboard) by issuing MIDI Program Change messages of the desired snapshot number.

When Snapshot Master for PlugIn is enabled, When Snapshot Master for Plug-In is enabled the instrument snapshots are available in the host program. There can only be one snapshot master. This setting can alternatively be activated for the entire ensemble (see ensemble properties.)

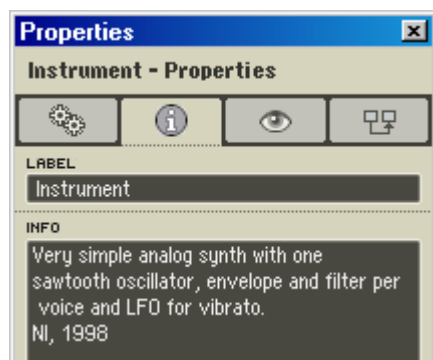
## Event Loops

When the **Event Loops** option is enabled, REAKTOR allows event-signal loops to occur within the instrument. These loops can lead to stack overflow crashes, which in turn can make the ensemble un-playable and, in some cases, un-openable.

When **Event Loops** is disabled, event-signal loops are prevented from occurring. If a loop is about to occur, REAKTOR displays a message revealing the source of the loop and asking you how to proceed.


We recommend that you disable **Event Loops** to maximize the stability of the instrument.

## 10.5.2. Info Page



Properties dialog of an instrument, Info page

Enter desired information about your instrument into the **Info** field of this page.

If  **Show Info** in the Ensemble Panel or Structure toolbar is enabled, your **Info** field text will be displayed in a popup whenever the mouse points to the instrument panel header.



## 10.5.3. Appearance Page

**Properties**

**Instrument - Properties**

**LABEL**

Instrument

**EDIT COLORS** **COLOR SCHEME**

Choose Color Select

ITEM	R	G	B
Panel	189	188	181
Indicator	255	160	0
Graph Line	255	213	53
Graph Fill	175	127	46
Graph BG	76	76	76

**STRUCTURE ICON** **PICTURE INDEX**

<none> ?

☒ Available in Panelsets

**ALL CONTROLS**

Visible Invisible

**BACKGROUND PICTURE** **PICTURE INDEX**

<none> ?

**PICTURE BORDERS**

BORDER TOP	BORDER LEFT
0	0
BORDER BOTTOM	BORDER RIGHT
0	0

Properties dialog of an instrument, Appearance page

### Edit Color

- **Choose Color:** Click this button to use the Color dialog palette to choose a color for the selected entry in the **Item** list below.
- **Color Scheme:** **Set to Custom** sets the instrument's color scheme to the current custom color scheme. **Save as Custom** saves the instrument's color scheme as the custom scheme. Note that REAKTOR only supports one custom color scheme; if you save a new scheme, you'll overwrite the previous scheme. **Set to Default** sets the instrument's color scheme to the default scheme (grey panel with orange indicators).

- **Item list:** This area lists all the instrument panel items that can have customized colors. You can use the Color dialog palette to choose colors for these items (see **Choose Color**, above). Or you can mix your own colors by using the **R**, **G**, and **B** fields to enter values for the color's red, green, and blue components. Each field accepts values from 0 (none) to 256 (full). Entering 0 for all three fields creates black; entering 256 creates white. And so on. Here are the color-customizable items:
  - Panel: Color of the panel background, if the panel has no background picture.
  - Indicator: Color of control (knob, fader, button, etc.) indicators.
  - Graph Line: Color of graph lines in tables, XY cursors, and fill outlines in filter and envelope displays.
  - Graph Fill: Color of graph fills in tables, XY objects, and fills in filter and envelope displays.
  - Graph BG: Color of the background of tables, XYs, and envelope and filter displays.
  - Grid: Color of the grid in table displays.
  - 2D Table Min: Color of the minimum value in the 2D table displays.
  - 2D Table Max: Color of the maximum value in the 2D table displays.
  - 2D Table Default: Color of the default value in the 2D table displays.

## Structure Icon

- **Structure Icon:** Lets you replace the default instrument structure icon (keyboard) with your own picture.
- **Picture Index:** If you choose a structure icon picture that contains multiple smaller pictures, you can select the index for the desired picture (after having set **Num Animations** in the Picture Properties dialog).

## Available in Panelsets

When **Available in Panelsets** is enabled, the instrument is included in the list at the bottom of the **Panelset** bar, and its panel can be shown (or hidden) in the Ensemble Panel window. When **Available in Panelsets** is disabled, the instrument is not listed in the Panelset bar, and its panel cannot be shown in the Ensemble Panel window.

## Panel Controls

- **A, B, AB:** These determine if changes you make to the appearance of the instrument are applied to panel A (**A**), panel B (**B**), or both panels A and B (**AB**). This affects two things: the **All Controls Visible** and **Invisible**

commands (see below) and the **Picture Borders** settings (see below). If **A** is enabled, the **All Controls Visible** and **Invisible** commands and **Picture Borders** settings will only be applied to panel A of the instrument. If **B** is enabled, the commands and settings will only be applied to panel B. If **AB** is enabled, the commands and settings will be applied to both panels, A and B.

- **Copy A > B, Copy B > A:** Click on one of these buttons to copy the full content and appearance of one panel to the other.

## All Controls

- **Visible** displays all of the instrument's controls in the panel(s) specified by **A**, **B**, and **AB** (see above).
- **Invisible** hides all of the instrument's controls in the panel(s) specified by **A**, **B**, and **AB**.

## Background Picture

- **Background Picture:** You can load your own picture for the instrument panel background. All panel controls and displays will lie on top of the background picture. You can assign a different background picture to each instrument panel (A and B).
- **Picture Index:** If you choose a background picture that contains multiple smaller pictures, you can select the index for the desired picture (after having set **Num Animations** in the Picture Properties dialog).

## Picture Borders

The **Border Top**, **Border Bottom**, **Border Left**, and **Border Right** values determine the number of pixels used for the top, bottom, left, and right borders of the instrument panel(s) specified by **A**, **B**, and **AB** (see above). Due to REAKTOR panel gridding, the Picture Borders values should be set to multiples of 4: 0, 4, 8, 16, etc.

## 10.5.4. Connection Page

**Properties** [X]

**Instrument - Properties**

[Settings] [Info] [Eye] [Link]

**LABEL**  
Instrument

---

**MIDI IN**

DEVICE: All CHANNEL: 1

UPPER NOTE G8	<input type="checkbox"/>	SUSTAIN CTRL 64
LOWER NOTE C-2	<input type="checkbox"/>	HOLD CTRL 66
NOTE SHIFT 0	<input type="checkbox"/>	MORPH CTRL 0

ALL NONE

---

**MIDI OUT**

DEVICE: All CHANNEL: 1

---

**OSC**

OSC SOURCE: no OSC Source TARGET: no Osc Target

**CONNECTIONS**

[Empty List] [Trash]

---

**AUTOMATION**

☒ Hide Name [IDS]

BASE ID: 0 MAX ID: 10 MAX ID IN USE: 10

Properties dialog of an instrument, Connection page

## MIDI In

- The **Device** drop-down menu specifies which of the available MIDI In devices the instrument can receive messages from. (You make MIDI In devices available to ensembles in the Audio Setup dialog, MIDI page.) Typically, **Device** is set to **All**, enabling the instrument to receive messages from all available MIDI In devices. In some cases, however, you might want to prevent an instrument from receiving messages from a certain MIDI In device.
- **Channel** specifies the MIDI Channel number used by the instrument for MIDI input. The instrument receives only those MIDI messages that are sent on the specified MIDI Channel number.
- **Upper Note** and **Lower Note** specify the range of MIDI In note numbers that the instrument will recognize. Any note numbers outside the range are ignored. This can be used to program a keyboard split.
- **Note Shift** enables all the MIDI In note pitches to be transposed up or down by the specified number of semitones. For example, if you want to transpose the whole instrument down by one octave you have to enter the value -12 here.
- **Sustain Ctrl** specifies the number of the MIDI controller that functions as a sustain pedal (called “hold” or “damper pedal” by some MIDI equipment manufacturers, standard controller number 64). As long as the sustain pedal is turned on, any playing note will be held, even after its key is released. To enable sustain for the instrument, turn on its **Sustain On/Off** option (the box left of **Sustain Ctrl**).
- **Hold Ctrl** specifies the number of the MIDI controller that functions as a hold pedal (called “sostenuto” by some MIDI equipment manufacturers, standard controller number 66). All notes that are playing when hold is turned on will be held even after their key is released until hold is turned off. Notes that are played when hold is already on are not affected. To enable hold for the instrument, turn on its **Hold On/Off** option (the box left of **Hold Ctrl**).
- **Morph Ctrl** defines the controller number used for the snapshot morph. The snapshot morph is activated by the button left to the controller field.

## All and None Menus

- Choose an option from the **All** drop-down menu to transfer it to all the instrument's controls.
- Choose an option from the **None** drop-down menu to remove it from all the instrument's controls.

To learn more about the **All** and **None** menu options, please consult the Panel Controls chapter.

## MIDI Out

- The **Device** drop-down menu specifies which of the available MIDI In devices the instrument can send messages to. (You make MIDI In devices available to ensembles in the Audio Setup dialog, MIDI page.)
- **Channel** specifies the MIDI Channel number used by the instrument for MIDI output.

## Connection

- The **OSC Source** drop-down menu specifies the OSC computer from which the ensemble receives MIDI data. Only computers which are present in the OSC member list in the **OSC Setup** dialog are available here.
- The **OSC Target** drop-down menu specifies the OSC computer to which the instrument sends MIDI data. Only computers which are present in the OSC member list in the **OSC Setup** dialog are available here.
- The **Connections** box lists the “pathname” of the instrument with respect to the internal structure of the ensemble. For example, if the instrument is named **Synth** and the ensemble is named **Ensemble**, the pathname will be displayed as **Ensemble/Synth**.

## 11. Primary Macros

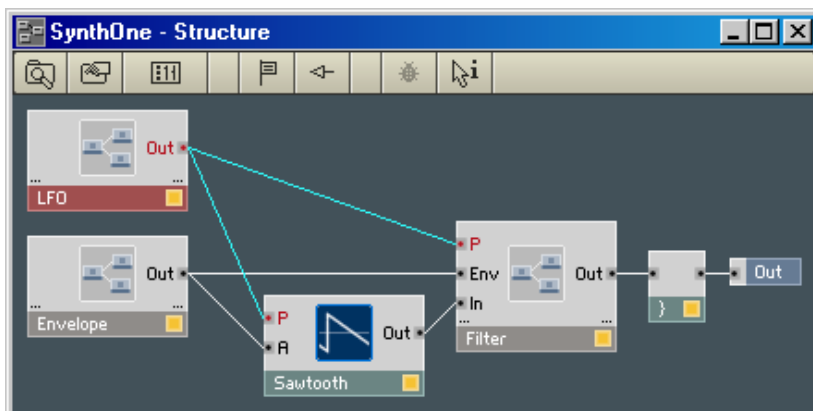
## 11.1. What is a Primary Macro?

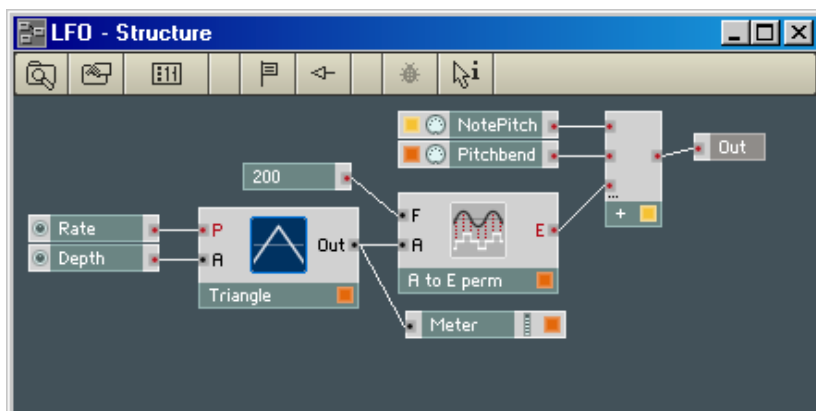
Primary macros have an internal structure just like instruments, but unlike instruments they have no MIDI management, separate panel, or snapshots. Primary macros have a gray label and can be recognized by the picture on their structure icon: 3 modules wired together.



## Macro object

The main application for primary macros is the encapsulation of functional blocks to obtain a hierarchical and clearer layout of complex structures. Extensive structures should always be realized using primary macros. Primary macros are also a convenient way to build re-usable components.





### Example for the integration of a macro into a structure

**Note:** To avoid wordiness, we will refer to “primary macros” as simply “macros” for the rest of this chapter. Bear in mind that what you read here applies to primary macros only, not to core macros.

## 11.2. Adding Macros to a Structure

You add macros to a structure by loading them from the REAKTOR system library or from your user content storage area (set in **Preferences** dialog, Directories page).

Use any of these methods to add a macro to a structure:



- XP: Right-click / OS X: Ctrl+click on a blank part of the structure, and use **Macro** from the context menu to find and insert the desired macro.
- Open the Browser (**View->Show Browser** or **F5**). Use the upper **Macro** button to find a system macro, or the lower **Macro** button to find a user macro (from your user content storage area). In the lower pane, drag the desired macro into the structure.
- Use the Browser's **Disk Navigation** controls (top row) to navigate to the desired macro folder, then drag the macro into the structure.

The system library provides a generous selection of premade macros. If you want to start developing a new macro, you first need to load an empty one (i.e. one that begins with **New**) from the system library.



When inserting a macro in a structure, you are in effect creating a copy of the macro file that is stored on your disk. The macro copy and the macro file are completely independent. Changes you make to the macro copy will not affect the macro file, and vice-versa. If you want to change the macro file, you must change the copy (in a structure), and then use **Save Macro As...** (from the macro's context menu) to save the changed copy over the existing macro file.

### 11.3. Ports

There is no fixed arrangement of input and output ports for macros. The type and number of ports in a macro is determined by the user by the insertion of ( ,  ) **terminals** in the instrument structure.

A terminal in the macro's structure appears as a port when you view the macro from its parent structure. For example, if you inserted one input terminal and one output terminal in a macro structure, and then double-clicked in the structure to move up to its parent structure, the macro icon would have an input port (left edge) and an output port (right edge). Thus, the signal enters the macro at its input port, gets processed inside the macro's internal structure, and is then sent back to the parent structure through the macro's output port.

---

**Note:** You can also create macro ports from within the macro's parent. Simply (XP) Ctrl+drag / (OS X): X+drag (i.e. hold down the Ctrl/X key while dragging the mouse) a wire from the desired port in the parent structure to the desired edge of the macro icon (left edge to create an input port, right to create an output port). When the new macro port appears, release the mouse button to create it. The macro port will get the name of the macro or module from whose port you Ctrl/X+dragged the wire.

---

## 11.4. Context Menu

The context menu of a macro contains these entries:

- **Mono**, when on, switches the macro to monophonic operation. For details, see Macro Properties, Function Page, Status below.
- **Mute**, when on, disables the selected macro. For details, see Macro Properties, Function Page, Status below.
- **Cut** removes the selected macro from the structure and stores it temporarily in the clipboard. From there, the macro can be pasted (using the **Paste** command) to a different structure (or another location in the same structure) .
- **Copy** does the same thing as **Cut**, but it does not remove the macro from the structure.
- **Duplicate** creates a copy of the selected macro in the same structure. Choosing **Duplicate** is equivalent to choosing **Copy**, then **Paste**.
- **Delete** deletes the selected macro from the structure.

---

**Note:** To save time, use the keyboard shortcuts for the above four commands: Cut = XP: **Ctrl+X** / OS X: **X+X**, Copy = XP: **Ctrl+C** / OS X: **X+C**, Duplicate = XP: **Ctrl+D** / OS X: **X+D**, Delete = **Del**. Also: Paste = XP: **Ctrl+V** / OS X: **X+V**.

---

- **Save Macro As...** enables the selected macro to be saved to an \*.mdl file on your disk. Builders typically use **Save Macro As...** to save new or modified macros to their user content Macros folder.
- **Structure** opens the selected macros' structure in the main structure window. Choosing **Structure** is equivalent to double-clicking on the macros' structure icon.
- **Structure Window** opens the selected macros' structure in a separate (i.e. not the main) structure window. Doing so enables multiple structure windows to be open at the same time. Choosing **Structure Window** is equivalent to Alt+double-clicking on the macros' structure icon.
- **Properties** opens the selected macros' Properties dialog. For details, see Macro Properties below.

## 11.5. Macro Properties

There are many ways to open a macro's Properties dialog. Use whichever one suits you best:

- Double-click on the macro icon's title bar (not its icon picture!) in a structure window.
- XP: Right-click / OS X: Ctrl+click on the macro icon in a structure window, and select **Properties** from the context menu.
- Select the macro icon in a structure window, and select **View->Show Properties** (or press **F4**).

Like all Properties dialogs, a macro's Properties dialog has four pages, each accessible from a picture button on the top of the dialog:



**Function,**



**Info,**



**Appearance, and**



**Connection.**

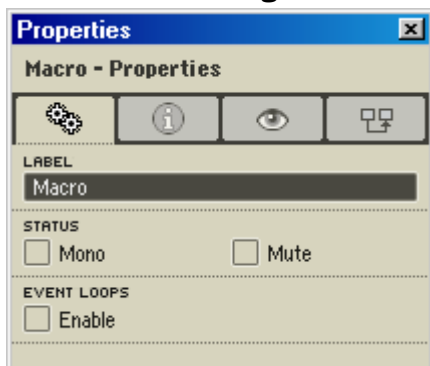
Note that its Connection page is empty.

---

**Note:** The contents of the Properties dialog automatically changes to display the values for the currently selected object (ensemble, macro, primary macro, etc.). So, to compare object values, leave the Properties dialog open and toggle your selection between the two objects.

---

### 11.5.1. Function Page



Properties dialog of a macro, Function page

## Label

The **Label** field specifies the name of the macro; i.e. the name that appears in the macro structure icon title bar and in the macro panel frame (assuming the macro has a frame). You can change this field to (re)name your macro.

## Status

- **Mono**, when on, switches the macro to monophonic operation by turning **Mono** on for all the modules inside. Since monophonic mode makes significantly less demands on your CPU, you should always turn **Mono** on, unless the macro must work polyphonically.
- **Mute** mutes (disables) the macro and all objects that lie upstream from it (i.e. that feed into it). In structure view, a macro whose Mute option is turned on has a red M over its status LED and red crosses over its input/output ports.

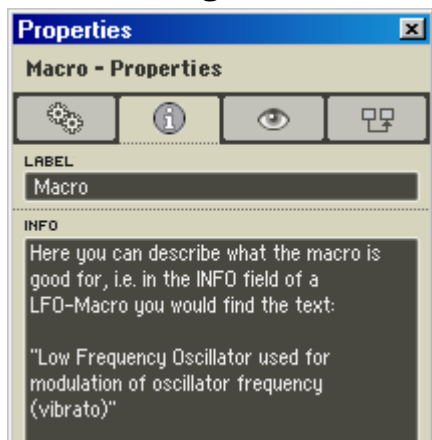
## Event Loops

When the **Event Loops** option is enabled, REAKTOR allows event-signal loops to occur within the macro. These loops can lead to stack overflow crashes, which in turn can make the ensemble un-playable and, in some cases, un-openable.

When **Event Loops** is disabled, event-signal loops are prevented from occurring. If a loop is about to occur, REAKTOR displays a message revealing the source of the loop and asking you how to proceed.


We recommend that you disable **Event Loops** to maximize the stability of the macro.

## 11.5.2. Info Page

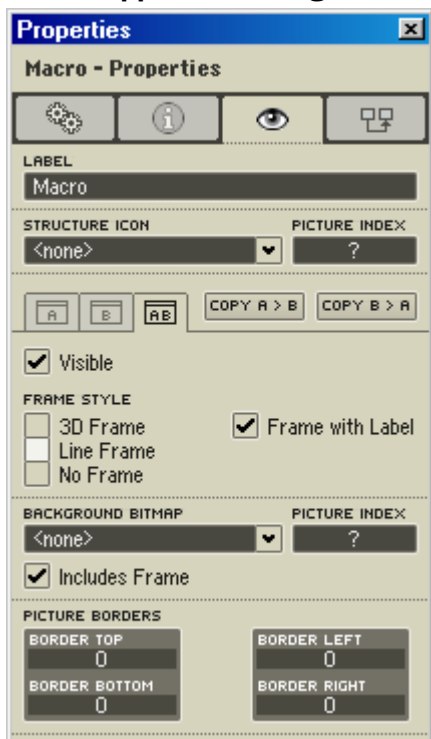


Properties dialog of a macro, Info page

Enter desired information about your macro into the **Info** field of this page.

If  **Show Info** in the Ensemble Panel or Structure toolbar is enabled, your **Info** field text will be displayed in a popup whenever the mouse points to the macro structure icon or panel frame (assuming the macro has a frame).

### 11.5.3. Appearance Page



Properties dialog of a macro, Appearance page

#### Structure Icon

- **Structure Icon:** Lets you replace the default macro structure icon picture (three modules wired together) with your own picture.
- **Picture Index:** If you choose a structure icon picture that contains multiple smaller pictures, you can select the index for the desired picture (after having set **Num Animations** in the Picture Properties dialog).

#### Panel Controls

- **A, B, AB:** These determine if changes you make to the appearance of the macro are applied to panel A (**A**), panel B (**B**), or both panels A and B (**AB**). This affects two things: the **Frame Style** options (see below) and the **Picture Borders** settings (see below). If **A** is enabled, the **Frame Style** options and **Picture Borders** settings will only be applied to panel

A of the macro. If **B** is enabled, the options and settings will only be applied to panel B. If **AB** is enabled, the options and settings will be applied to both panels, A and B.

- **Copy A > B, Copy B > A:** Click on one of these buttons to copy the full content and appearance of one panel to the other.

## Frame Style

- **3D Frame:** Displays a 3D frame around the macro controls in panel view.
- **Line Frame:** Displays a line frame around the macro controls in panel view.
- **No Frame:** Displays no macro frame in panel view.
- **Frame with Label:** Displays the label (name) of the macro in its panel frame.

## Background Bitmap

- **Background Bitmap:** You can load your own picture for the macro panel background. All panel controls and displays will lie on top of the background picture. You can assign a different background picture to each macro panel (A and B).
- **Picture Index:** If you choose a background picture that contains multiple smaller pictures, you can select the index for the desired picture (after having set **Num Animations** in the Picture Properties dialog).
- **Includes Frame:** Displays a frame around the background picture.

## Picture Borders

The **Border Top**, **Border Bottom**, **Border Left**, and **Border Right** values determine the number of pixels used for the top, bottom, left, and right borders of the macro panel(s) specified by **A**, **B**, and **AB** (see above). Due to REAKTOR panel gridding, the Picture Borders values should be set to multiples of 4: 0, 4, 8, 16, etc.

## 12. Primary Structures

### 12.1. What is a Primary Structure?

REAKTOR is based on an open concept that allows for the design and realization of any imaginable sound generator. In many respects it is similar to a classic modular synthesizer system. That's why the most important basic building block you deal with in REAKTOR is called a **module**. (Primary module in the primary level; core module in the core.)

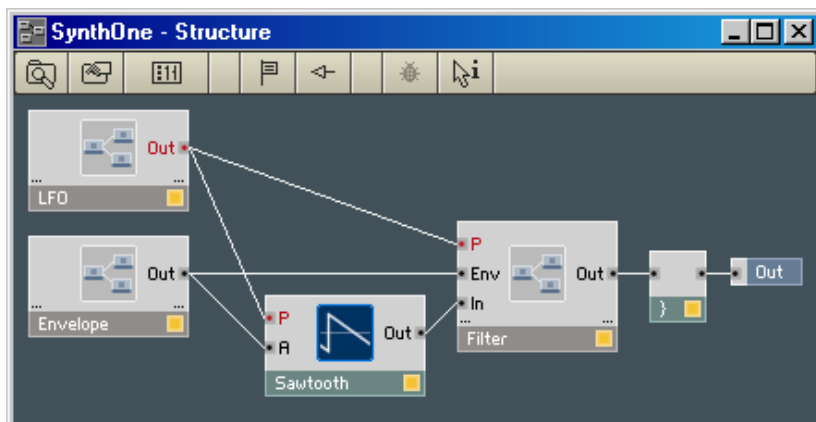
A library of primary (and core) modules is built into REAKTOR. These built-in modules provide the basic building blocks for MIDI and audio signal processing. Complex signal processing structures can be created by connecting modules that carry out relatively simple tasks.

The window in which primary modules are placed and interconnected is called a **Primary Structure window**.

---

**Note:** To avoid wordiness, we will refer to “primary structures, macros, and modules” as simply “structures, macros, and modules” for the rest of this chapter. Bear in mind that what you read here applies to primary structures, macros, and modules only, not to core structures, macros, and modules.

---



A Structure window



We strongly recommend that you keep to hierarchical principles when building structures in REAKTOR. The ensemble should (and, in fact, can) contain instruments only. Instruments should contain macros and modules and core cells only (not other instruments). Macros should contain other macros and modules and core cells (not instruments).

When creating complex devices, it is important to maintain a clear layout. The following recommendations will help you maintain an appropriately clean design.

- Only instruments, not macros or modules, can reside in an ensemble structure window. To this end mixers, which you use to mix signals from several instruments, are available as instruments in the system library.
- During the construction of instruments, group as many functional blocks as possible in the form of macros. One advantage to working this way is that identical elements (e.g., oscillators and envelopes), which are often used more than once in the construction of synthesizers, need only be constructed once and can then be copied as needed. Also, your structures will be very clear, which makes tracking down problems much easier.

## 12.2. Modules

A module is the smallest hierarchical unit in REAKTOR. It is displayed as a graphical object. Each module is marked with a **label** and a **picture icon** (e.g., oscillators have waveform pictures).



The Pulse FM oscillator module

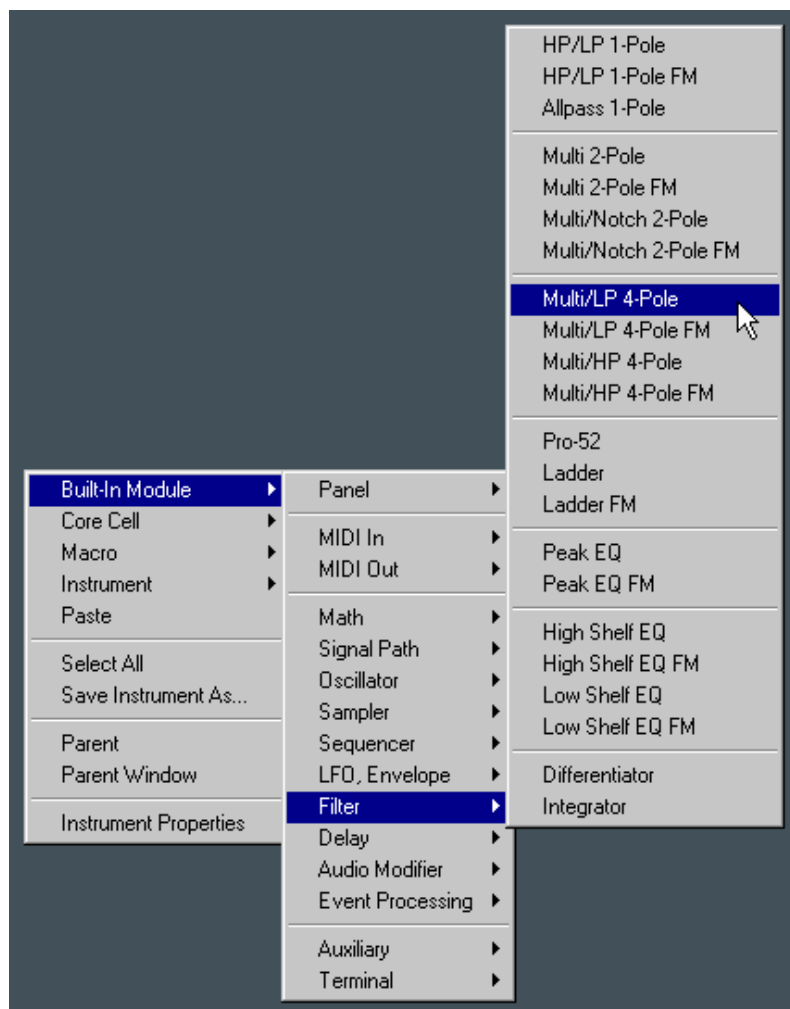
### Adding Modules to a Structure

To add a new module to a structure, use the structure window's context menu. The submenu **Built-In Module** lets you select a built-in module from REAKTOR's system library. A popup menu with several levels appears:

First you must select the functional group (e.g., **Filter**) and then choose the actual module that belongs to the group (e.g., **Multi/LP 4-Pole**). Detailed

information about all of REAKTOR's modules can be found in the Module Reference section of this manual.

The module icon will be placed at the point in the structure window where you opened the context menu (by right-clicking or Ctrl+clicking), but you can move it around like any other REAKTOR object.



Menu for inserting a new module

## Module Ports

Each REAKTOR module contains one or more ports through which the module can be connected to other modules. The left edge of the module holds the

122 – REAKTOR 5

input ports and the right edge holds the output ports.

When any input port is left unconnected, it receives a zero (0) signal. So, connecting no wire to an input port has the same result as connecting a constant source with the value set to 0.

REAKTOR distinguishes between two kinds of information that can be understood or sent by a port, **audio** and **event**:

- **Audio** signals are comparable to sound signals and control voltages in the analog world. The processing of such a signal constitutes a permanent load on the CPU. Ports for audio signals are labeled with black characters. When wiring audio ports, note that an audio input can never process more than one signal. If two or more audio signals are to be fed to an audio input, they must first be mixed using an **Add** or **Amp/Mixer** module. If a connection is made to an audio input port that already has a wire attached, the first wire will be deleted as soon as the second one is connected.
- **Event** signals are control messages for changing a value. Typical sources for events are MIDI inputs and panel faders. Since event processing allows complex manipulation of control messages without continuous calculations, the load on the CPU is reduced. Ports for event signals are labeled with red characters and marked with a small red dot. If two or more event signals are to be fed to an event input, they must first be merged using a **Merge** module. Gate signals are a special type of event signal. An event with a non-zero value turns on the gate. When it is followed with a zero (0) event, the gate is turned off again.

Some modules can be used either for audio or event signals. If you insert such a module (e.g. the **Add** module), it will appear first as an event module (i.e. its ports will be red). As soon as you connect an audio wire to one of its inputs, however, it will convert to an audio module and the CPU load will become substantially higher with each additional connection.

Each port has a **context menu** with the following entries:

- **Create Control** automatically creates a suitable panel controller for the port (see section **Panel Editing** and **Panel Operation** for details about working with controls on the panel).
- **Create Constant** automatically creates a Constant module with a suitable value for the port.
- **Mute Port** mutes the port (i.e., sets its value to zero). Muted ports are marked with a red cross.

## Module Context Menu

### Mono

A module can operate either in monophonic (single-voice) mode or in polyphonic (multiple-voice) mode. In polyphonic mode, processing is carried out for several voices in parallel. The number of voices of a polyphonic module is determined by the instrument to which the module belongs. Polyphonic modules can be identified by the yellow color of the status LED at the bottom left corner of the module. Monophonic modules have an orange status LED.

For most modules the operating mode can be changed using the entry **Mono** in the context menu or the **Mono** option in the module's Properties dialog (Function page). Unless a module really needs to be run in polyphonic mode, it should always be used in mono mode, since the CPU load increases proportionally to the number of voices used.

### Mute

A module can be muted by selecting **Mute** from its context menu or Properties dialog (Function page). Muted modules are recognized by a red cross over the status LED.

A muted module makes no computational demands on your CPU. If a module is not needed temporarily, it should be muted; if it is never used, it should be deleted.

Modules are automatically disabled if their outputs are not connected or are only connected to other disabled or muted modules. The status LEDs of disabled modules are unlit.

This feature is especially useful with switches, because alternative branches of signal processing can be selected but only one will cause CPU load. It works like this: Only one of the inputs of a switch is active at any one time – the switch position determines which input. The signals of all the modules connected to inactive inputs are therefore not needed. REAKTOR turns them off, so that they do not cause any unnecessary load on the CPU.

### Cut, Copy, Duplicate

- **Cut** removes the selected module from the structure and stores it temporarily in the clipboard. From there, the module can be pasted (using the **Paste** command) to a different structure (or another location in the same structure) .
- **Copy** does the same thing as **Cut**, but it does not remove the module from the structure.

- **Duplicate** creates a copy of the selected module in the same structure. Choosing **Duplicate** is equivalent to choosing **Copy**, then **Paste**.

## Delete

**Delete** deletes the selected module from the structure.

---

**Note:** To save time, use the below keyboard shortcuts for the above four commands:

---

- Cut: XP: Ctrl+X / OS X: X+X
- Copy: XP: Ctrl+C / OS X: X+C
- Duplicate: XP: Ctrl+D / OS X: X+D
- Delete: Del
- Paste: XP: Ctrl+V / OS X: X+V

## Properties

A dialog with information about the macro can be opened with the context menu entry **Properties**. For detailed information about all modules please see the **Module Reference** section of this manual.

## 12.3. Source Modules

### What are Source Modules?

In REAKTOR, **source module** is the name given to a module that outputs a control signal. There are three different kinds of source modules:

- **Control source modules** have a representation on the panel. The panel element is used to set the value of the control signal.
- **MIDI source modules** convert MIDI data to control signals.
- **Constant source modules** have a fixed value.

## Control Source Modules

The **Fader**, **Knob** and **Button** modules are examples of control sources. There are two ways to insert them into a structure:

- Choose the desired module from the context menu of the structure window (**Built-In Module** ⇒ **Panel** ⇒ **Fader / Knob / Button**).
- In the context menu of a module input port select **Create Control**. A control source is created and connected to the input. Type, label, and settings of the control source are configured to suit the input; note that you might have to change these to fix your needs. In many cases you can save a lot of time by using **Create Control** to add control sources.

Control sources and their respective panel elements can be controlled via MIDI in various ways.

Control sources have their own context menu which you open by XP: right-clicking / OS X: Ctrl+clicking on the module. The menu choices include: **MIDI Learn**, **Cut**, **Copy**, **Duplicate**, **Delete** and **Properties**.

## MIDI Source Modules

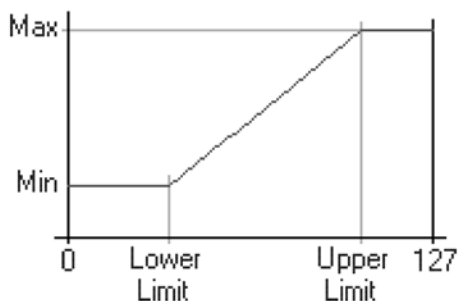
MIDI source modules are used for controlling audio signal processing with MIDI events. For each type of MIDI event there exists a particular kind of source module. The output signal of such a source corresponds to the values transmitted by the particular MIDI events. For example, the **On Vel.** source module outputs a control signal that corresponds to the Note On Velocity message transmitted by MIDI when a MIDI keyboard key is pressed.

MIDI source modules are inserted through the context menu of the structure window by choosing **Built-In Module** ⇒ **MIDI In...**

## Range of Values

For control and MIDI source modules the range of the output control signal is scaled to the range between **Min** and **Max** (as set in the module's Properties dialog, Function page) to achieve optimum control of the particular module parameter.

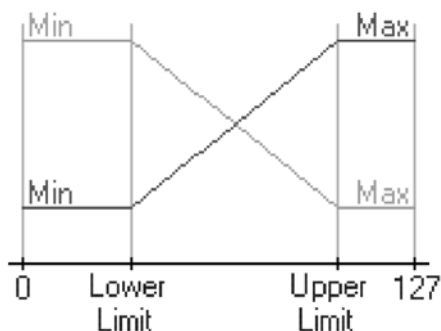
For MIDI source modules the range can also be limited with the Properties **Lower Limit** and **Upper Limit**. The output value of the source module is limited to **Min** for MIDI values below **Lower Limit** and to **Max** for MIDI values above **Upper Limit**. The range between the two limits is interpolated linearly between **Min** and **Max** as shown on the diagram:



### Scaling and limiting

The values for **Lower Limit** and **Upper Limit** lie between 0 and 127 and the value for **Upper Limit** must be greater than that set for **Lower Limit**.

However, **Max** can be smaller than **Min** to achieve inverted operation. If opposite characteristics are set for two sources, a crossfade effect can be programmed:



### Crossfade

A switch with adjustable threshold level can be emulated by setting **Lower Limit** and **Upper Limit** to neighboring MIDI values, e.g. 63 and 64. When the input value to such a source is below 64 **Min** is output, otherwise the output value is **Max**.

### Stepsize

The range of values in source modules normally has a resolution of 128 steps. In many modules (particularly **Fader** and **Knob**) the parameter **Stepsize** can be used to reduce the resolution to fewer than 128 steps. You enter the step size by which the output value is to change, beginning at **Min**. For example, you can set a pitch parameter to select only octaves by giving it a **Stepsize** value of 12.

## Constant Source Modules

Module and macros with fixed values need to be fed by constant source modules. Set the desired **Value** in the Properties dialog (Function page) of the **Constant** module.

To insert a constant source module, select **Built-In Module** ⇒ **Math** ⇒ **Constant** in the context menu of the structure.

## 12.4. Switches

Switches are not source modules because they do not generate any control signals. They are controls, however, because (like other control sources) they are represented on the panel by a control element.

Several modules or macros can be connected to the inputs of a switch and the position of the switch then determines which signal is passed to the output of the switch. An exception are switches of type “1” which only toggle between activating and deactivating the signal path. They are simply on/off switches for signals. Details on switches can be found in the Module Reference section of this manual.

The use of switches in a structure can also play a significant part in reducing the load on your CPU. This is because modules or parts of the structure that are not connected to REAKTOR's audio outputs (or to the input of a Tapedeck module) do not add anything to the audio signal and are therefore automatically switched off. In this state they do not cause any CPU load. For example, you may use a switch to select one of several oscillators. Only the oscillator whose signal is being output will be active, while all the other oscillators are automatically deactivated.

## 12.5. Terminals

Terminals are very inconspicuous but immensely important modules in REAKTOR structures. They are like the sockets on hardware instruments. Each input or output terminal within a structure appears at the next higher level – i.e., in the instrument or macro – as a port from which connections to other instruments, macros and modules can be made.

According to the different kinds of module ports, several types of terminal ports are available: **In Port**, **Out Port**, **Send**, **Receive**, **IC Send**, **IC Receive**, **OSC Send** and **OSC Receive**. The normal rules for wiring apply.

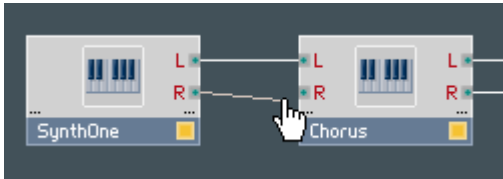
Terminals are created using the context menu of a structure window under **Built-In Module** ⇒ **Terminal....** The **Label** of an In Port or Out Port terminal is initially just **In** or **Out**, but if you have several Ins or Outs you should give them



meaningful names (like **L** and **R** in the following picture) to prevent any possible confusion. You should also provide terminals with a description (Properties dialog, Info page). The terminal label appears as the port label in the parent structure, and the terminal description appears in a popup info box when the mouse points to the port (if **Show Info** is enabled).

## 12.6. Wires

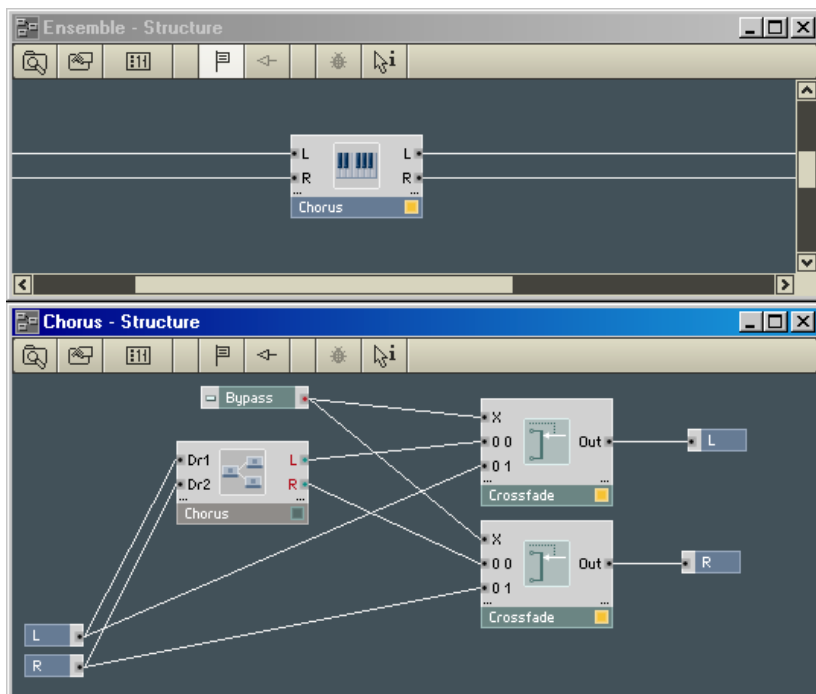
The connection between the ports of two modules or macros, shown as a line, is called a **wire**. Wires transport signals between the modules/macros.



Connecting a wire

## Creating

To make a new wire:



Instrument with ports and its structure with terminals

- Click on one of the two ports to be connected with the left mouse button, drag the mouse pointer to the other port, and then release the mouse button. A wire appears between the ports and the effect on the sound resulting from the change to the structure can be heard immediately.

## Deleting

There are two ways to delete a wire:

- Do the same as you would to create a new wire; i.e. drag from one port to the other.
- Drag the mouse from the input port to which the wire is connected to a blank part of the structure.
- Select the wire you want to delete by clicking on it, then press the **Del** key on your computer keyboard.

## Rules for Wiring

When wiring modules together, a few **general rules** always apply:

- A wire can only connect an output port to an input port or vice-versa. It can never connect an input port to an input port, or an output port to an output port.
- An output port can feed (be connected to) as many as 40 input ports.
- When no wire is connected to an input port, it receives a zero (0) signal, meaning that the value at the port is 0.

In addition, the following special rules apply:

- An event input port cannot process audio signals. If an event input port is to be fed from an audio output port, the signal must first be converted with an **A to E** module (see the **Module Reference** section of this manual).
- An event output port can be connected to audio input ports as well as event input ports.
- When connecting a monophonic signal to a polyphonic input port, all polyphonic voices receive the same value (from the monophonic signal). For pitch signals this means the voices in effect play in unison.
- A polyphonic output cannot be connected to a monophonic input (a red cross appears on the input port). An **Audio Voice Combiner** module must be used for converting poly to mono.

## Displaying Wire Signal Values

While the mouse pointer rests on a wire (and if **Show Info** is on), the value of the signal in the wire is shown in a popup info box.

For event signals, the value of the last event is shown. (If events come faster than the rate at which the display updates, some intermediate values may be missed.)

For audio signals, a rough indication of minimum and maximum values, i.e. the range of the signal, is given. (Short peaks in the signal may be missed and thus do not show up in the display). If the range of the signal is constantly changing, you may need to move the mouse pointer away from the wire to close the popup, and then point on the wire once more to start measuring minimum and maximum values again.

For polyphonic signals, the values for all voices are displayed with one line of values for each voice. At the left, the MIDI note numbers which are playing

on the respective voices are shown. If a voice is not playing any note, **Note: Off** is displayed along with the value of the signal on the wire. Voices with note off are always shown below the voices with note on.

## 12.7. Signal Processing in REAKTOR

REAKTOR distinguishes two kinds of signals: event and audio. Event signals are typically processed at a rate of several hundred times per second, whereas audio signals are processed at the audio sampling rate, which is tens of thousands of times per second. For example, the standard audio sampling rate for Compact Discs is 44,100 times per second (usually written as 44.1 kHz). Having two processing rates conserves CPU load. Both the sample rate and the control rate (which is used by a handful of modules), can be changed from REAKTOR's **Settings** menu.

REAKTOR's audio generating and processing modules process signals at the audio rate. There are a few modules in REAKTOR, such as **Event Smoother**, **LFO**, **Slow Random** and **A to E**, that generate and process event signals at the control rate.



Even Smoother module



LFO module



Slow Random module



A to E module

However, some event modules do not scan continuously for events, but only react when a new event arrives. A new event can be created from inside the structure, by a mouse action (i.e., when you move a panel control with the mouse), by an incoming MIDI message, or even by an audio event. When an audio signal is used to create event signals (e.g., using the **A to E Trig**) an event output port can even produce a signal which is refreshed at the audio rate.

Event input ports compute all incoming events regardless of their rate. A special case is the **Iteration** module, which can compute even multiple events within one audio sample. Finally, there are hybrid modules that can be configured to process signals at either rate - math modules are a typical example. On those modules, the ports are marked with three different colors to indicate their mode:

- A green dot on a port of a hybrid module indicates that the mode has not yet been set and you can connect either an event or audio signal.
- A red dot on a port of a hybrid module indicates that the mode has been set to **event** by connecting an event cable to the module.
- A black dot on a port of a hybrid module indicates that the mode has been set to **audio** by connecting an audio cable to the module.

## Event Signals

Event signals are control messages for changing a value. Typical sources for events are MIDI inputs and panel faders. Event processing allows for complex manipulation of control messages without continuous calculations and thereby reduces the load on the CPU in comparison to the calculation demands of audio signals. Ports for event signals are labeled with a red dot and a red label. To connect more than one event cable to an event input port, place a **Merge** module in front of it. An audio output port cannot be connected directly to an event input port; you must use an **A to E** converter module for this purpose.

Gate signals are a special case of Event signals. An event with a non-zero value turns on the gate. When it is followed with a zero- or negative-valued event, the gate is turned off again.

An event has two properties: the time at which it occurs, and the value it carries, which is the new value when used as an audio signal.

Every event signal is also an audio signal, so it has a value for every sample. The difference is that this value is constant, until an event comes along to change the value. This means that every event output can also be used like an audio output, but the signal will be stepped, not smooth.

Some modules (**A to E**, for example) only evaluate an audio signal connected to its input at the control rate.

Most modules that operate on Events (e.g. **Add** used as event module) work at the exact moment an event arrives (i.e., event timing keeps perfect pace with the audio sample rate). Other event modules (**A to E** or **LFO**, for example) operate only at the lower timing resolution determined by the Control Rate (e.g., 200 times per second).

## Order of Event Processing

Most event processing modules, in response to an input event, generate an output event immediately. That is, an event travels through the chain of event modules to the end (possibly fanning out if there are several paths) before the next event travels the chain.

The method is called “depth before breadth”: An event propagates as deep as it can along one track before another wire fanning out from the same port is processed.

If one event is to go down more than one branch, and you need to have the branches executed in a defined order, you should use the **Order** module to fan out to the different paths.

Another important module in this context is the **Value** module. A complex event processing structure can be connected to its lower (value) input, but it will only pass on this value as an event when a triggering event arrives at the Trig input. You can look at this as a Sample&Hold circuit triggered by an event. You can use the **Order** module to generate this triggering event and make sure that it occurs after other event processing has completed.

When different source modules produce events at the same moment in time - for example, when they are initialised as soon as the structure is turned on - they actually send events in the order in which the source modules were originally inserted into the structure. To have one module initialised after the others, simply cut it and paste it back into the structure.

## Event Loop Prevention

The **Globally disable event loops** option in the **Preferences** dialog (Options page) and the **Event Loops Enable** options in the ensemble/instrument/macro **Properties** dialogs (Function page) allow the suppression of event signal loops which can lead to stack overflow crashes.

---

**Note:** Unprotected **Event Loops** will crash REAKTOR. This is not bug, but by design. This is only avoidable by careful instrument design. The fuse for this case is often the insertion of a **Value** module.

---

Disabling event loops (by turning on **Globally disable event loops** or turning off **Event Loops Enable**) prevents event-signal loops from occurring in ensembles. If an event loop is about to occur, REAKTOR displays a message revealing the source of the loop and asking you how to proceed.

Event loops can lead to stack overflow crashes, which in turn can make ensembles un-playable and, in some cases, un-openable. If this happens, restart REAKTOR, turn on **Globally disable event loops**, open the problematic ensemble, and trace the source of the event loop with the help of event-loop identification messages. (It can be useful to disable audio to prevent further loops occurring during this process.)

We recommend that you globally disable event loops to maximize the stability of REAKTOR. To ensure backward compatibility, files saved in older versions of REAKTOR have event loops enabled by default.

---

**Note:** In most cases, the **Iteration module** can avoid the need for creating event loops. The Iteration module has a limited speed option in its properties, which can avoid audio glitches caused by processing a large number of iterations.

---

## Audio Signals

Audio signals are comparable to sound signals and control voltages in the analog world. The processing of such signals constitutes a permanent load on the CPU. Audio module ports for audio signals are labeled with black characters. When wiring audio ports, note that an audio input can never process more than one signal. If more than one audio signal is to be fed to an audio input, they must first be mixed using an **Adder** or **Amp/Mixer** module. If a connection is made to an audio input port that already has a wire attached, the first wire will be deleted as soon as the second one is connected.

## Activating Audio modules

Since audio modules are a constant drain on the CPU, REAKTOR automatically disables modules (both audio and event) that are not ultimately connected to the audio output. By “ultimately” we mean connected by some series of wires from the modules output to the ensemble’s Audio Out module (of which there is always and only one). You can tell that a module is active by the glowing LED in its lower-right corner.

Some audio modules (lamps for example) can be set to be always active in their Properties dialog (Function page). In that case, they will always be active (notice that their LED comes on when this property is set for an unconnected module), and they will activate any modules connected to them. Modules with the Always Active option have another special property that doesn’t depend on the Always Active option being turned on: when any of their input ports are connected, they will “look back through the signal path” to see if there

are any active modules connected to them, and if so, they will become active. Lamps are a good example of the reason for that - they have no outputs to make them active, but you would want them to be active whenever they are connected to an active module (i.e., whenever there's something for them to indicate).

## Order of Audio Processing

Unlike event processing, whose order depends on the order in which they were created, audio modules are processed in an order that depends on their position in the signal flow. You can see the processing order of audio modules in the structure by selecting **System->Debug->Show Module Sorting** from the main menu.

The sorting process is relatively straight forward until a feedback loop is encountered. Feedback loops are allowed - they are useful for physical-modeling instruments, for example - but REAKTOR must arbitrarily assign a sorting order for such paths. The first module in a feedback loop is indicated by a vertical blue line at the appropriate port. This indicates an automatically inserted **Unit Delay**. This **Unit Delay** is not visible since it resides inside of the feedbackin module. You can also set the starting point manually by making use of the **Unit Delay** module.

## 12.8. Context Menu

The context menu of the Structure window has the following entries:

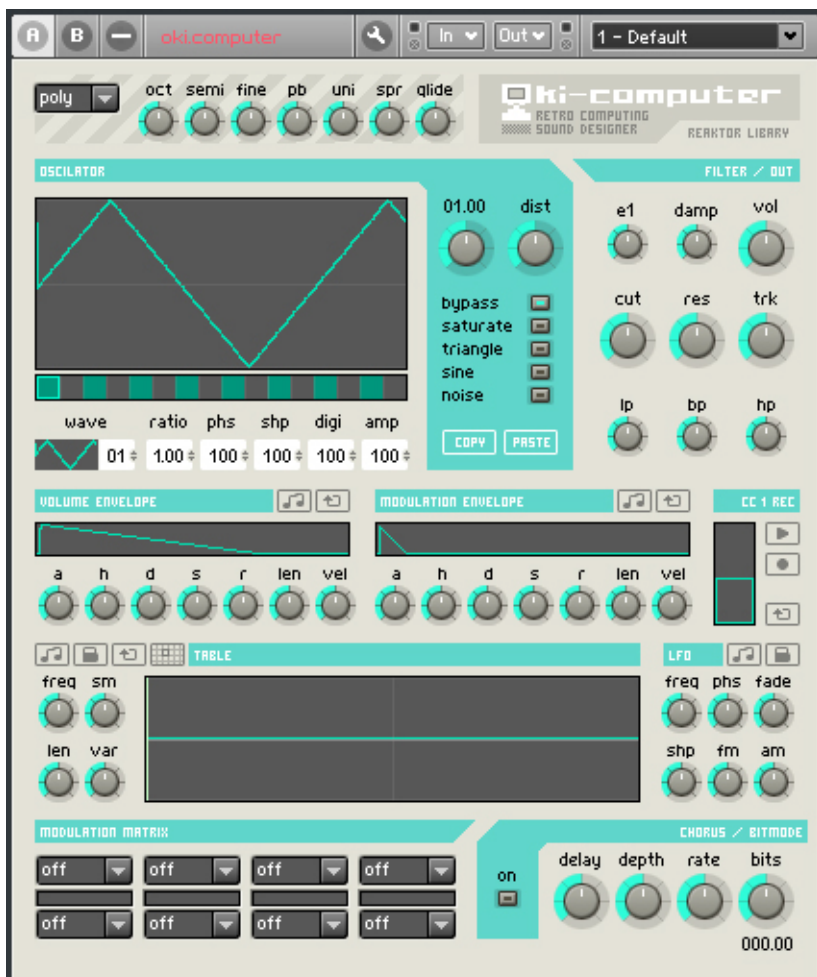
- **Built-In Module** inserts modules into the structure.
- **Core Cell** inserts core cells into the structure.
- **Macro** inserts macros from the library into the structure.
- **Instrument** inserts instruments into the structure.
- **Paste** inserts a previously cut or copied object into the structure at the point where the context menu was opened. When using the keyboard shortcut Windows XP: **Ctrl+V** / OS X: **X+V** for pasting, you can specify a point in the structure by clicking on it with the left mouse button first.
- **Select All** selects all the objects in the structure.
- **Save Instrument/Macro As...** saves the structure to a file with a new name. Depending on the type of structure (instrument or macro) the correct filename extension will be appended (.ism or .mdl).



- **Parent** opens the parent structure in the same structure window. For example, if you are in the structure of a macro which resides in an instrument, **Parent** will open the instrument's structure in the same window.
- **Parent Window** opens the parent structure in a separate structure window.
- **Instrument/Macro Properties** opens the Properties dialog of the structure's instrument or macro.

# 13. Panel Editing

## 13.1. What Is a Panel?

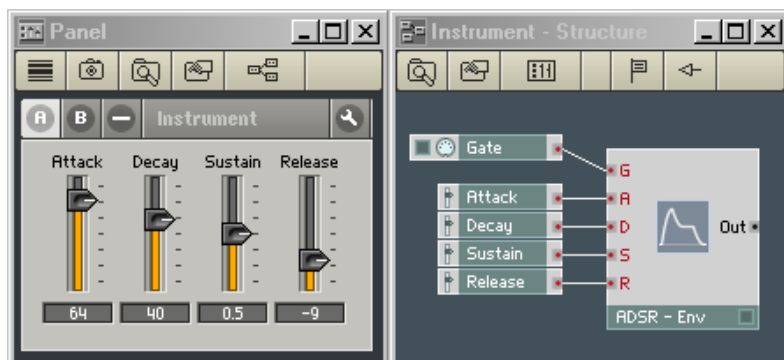


Ensemble Panel window of the OKI Computer ensemble from the REAKTOR 5 factory library

A **panel** is the user interface of an instrument. It corresponds to the front panel of a hardware synthesizer or effects unit, where the various elements (knobs, faders, buttons, meters, etc.) for operating the device are located. Instrument panels are displayed in the Ensemble Panel window.

## 13.2. What are Panel Controls?

Some REAKTOR modules generate or modify audio signals (oscillators, filters, samplers, saturators, etc.). Others control the signal flow by enabling different values to be sent to module inputs (knobs, faders, buttons, etc.). When these control modules are displayed in an instrument panel, they are called **panel controls**.



On the left a panel with faders, on the right the structure with the corresponding source modules

## 13.3. Panel Controls

In this section, we'll look at five of REAKTOR's most commonly used panel controls: faders, knobs, buttons, switches, and lists.

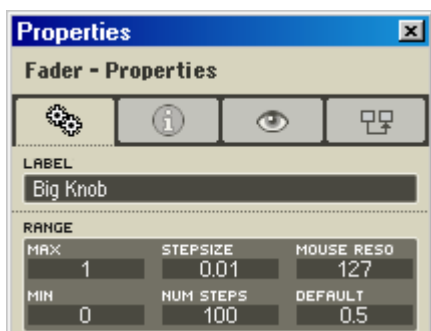
### Fader and Knob



Different types of faders and knobs

Faders and knobs are panel controls whose settings (i.e. fader-handle and knob-indicator positions) determine the values that their source modules (Fader and Knob) output to other modules in the structure (e.g. the P input of

an oscillator, or the A input of a sampler). Their output value range is set by **Min** and **Max** in their Properties dialog (Function page). Their step resolution (the number of increments between Min and Max) is set by **Stepsize**, and their mouse resolution (the distance the mouse must travel to change the knob/fader settings) is set by **Mouse Reso**.



*Properties dialog of a knob (Function page)*

---

**Note:** If you set **Stepsize** to 0, REAKTOR will automatically change it to a value that yields a total of 127 steps between **Min** and **Max**. This resolution suffices for most controls.

---

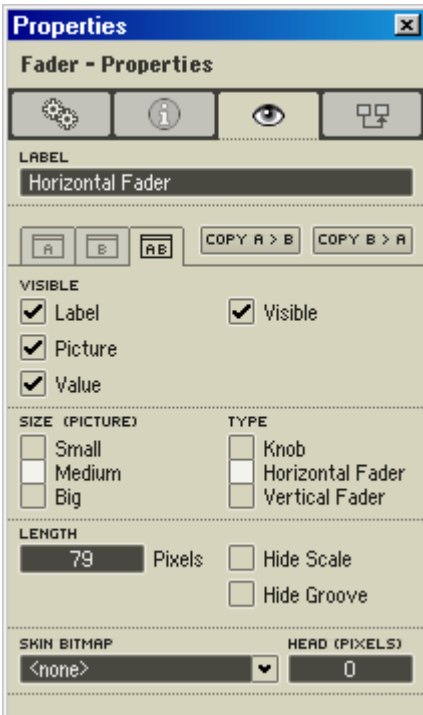
You can change the fader/knob panel setting by dragging over it with your mouse, or by pressing your Up/Down arrow keys (if you click the fader/knob first to select it). You can also use MIDI to change fader/knob settings (see below, MIDI Control).

---

**Note:** Drag your mouse up and down (not sideways!) to change a knob setting.

---

You can change a fader's/knob's panel appearance in its Properties dialog (Appearance page):



*Properties dialog of a fader (Appearance page)*

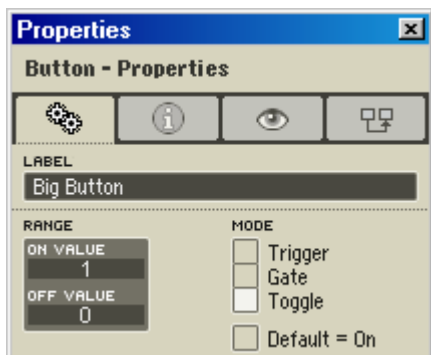
- **Visible (Label, Picture, Value, Visible)** - **Label** shows/hides the label in the panel, **Picture** shows/hides the fader/knob graphical image, and **Value** shows/hides the current (output) value. **Visible** shows/hides the entire fader/knob (label, picture, and value).
- **Size (Small, Medium, Big)** - determines the size of the fader/knob panel display.
- **Type (Horizontal Fader, Vertical Fader, Knob)** - determines the panel display type. Note that you can display a Fader module as a knob in the panel, and a Knob module as a fader.
- **Length** - the length (or height) of the fader in pixels. This has no effect for knobs.
- **Hide Scale, Hide Groove** (faders only) - shows/hides the fader's scale tickmarks and groove (the slot in which the handle fits).
- **Skin Bitmap, Head** - see below, Panel Control Skins.

## Button



### Different types of buttons

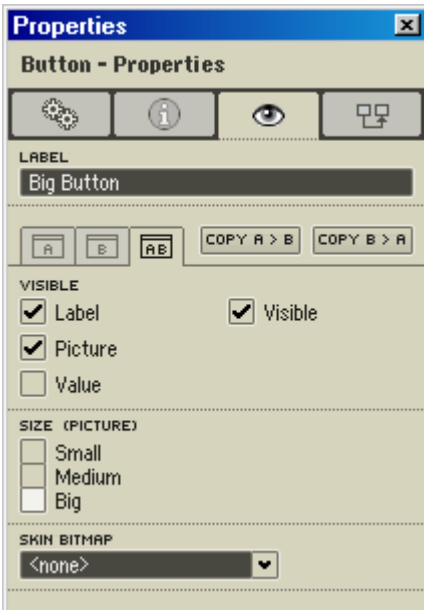
A button is a panel control whose setting (on or off) determines the value that its source module (Button) outputs to other modules in the structure (e.g. the G input of a sampler, or the A input of a clock oscillator). Its output value range is set by **On Value** and **Off Value** in its Properties dialog (Function page).



### Properties dialog of a button (Function page)

You turn a button on/off by clicking it with your mouse. You can also use MIDI to turn buttons on/off (see below, MIDI Control).

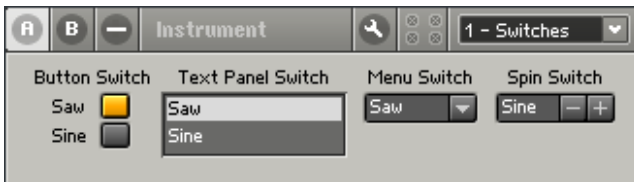
You can change a button's panel appearance in its Properties dialog (Appearance page):



Properties dialog of a button (Appearance page)

- **Visible (Label, Picture, Value, Visible)** - **Label** shows/hides the label in the panel, **Picture** shows/hides the button's graphical image, and **Value** shows/hides the current (output) value. **Visible** shows/hides the entire button (label, picture, and value).
- **Size (Small, Medium, Big)** - determines the size of the button panel display.
- **Skin Bitmap** - see below, **Panel Control** Skins.

## Switch



Different types of switches

A switch is a panel control whose setting (selected option) determines which of its source module's input signals is passed to its output port. For example, you might have a switch that receives two input signals, one from a sawtooth

oscillator and the other from a sine oscillator. If the Sawtooth option in the switch is turned on, the sawtooth input is passed to the switch output; if the Sine option is on, the sine input is passed to the output.

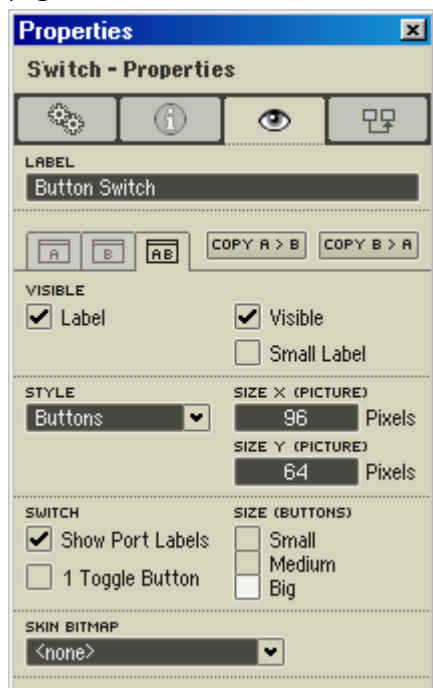


### Switch module

A switch can have multiple inputs (as in the sawtooth/sine example) or a single input. In a single-input switch, the switch setting determines whether the input signal is or is not passed to the output.

Along with your mouse, you can use MIDI to change switch settings (see below, **MIDI Control**).

You can change a switch's panel appearance in its Properties dialog (Appearance page):



Properties dialog of a switch (Appearance page)

- **Visible (Label, Small Label, Visible)** - **Label** shows/hides the label in the panel, **Small Label** shows/hides a small version of the label. **Visible** shows/hides the entire switch.



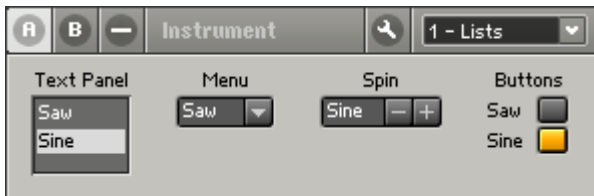
- **Style (Buttons, Menu, Text Panel, Spin)** - **Buttons** displays switch options as buttons, **Menu** displays them as text items in a drop-down menu, **Text Panel** displays them as text items in a box, and **Spin** displays them as text items in a menu with +/- navigation buttons.
- **Size X, Size Y** - specify the width and height (in pixels) of a switch whose style is set to Menu, Text Panel, or Spin.
- **Switch (Show Port Labels, 1 Toggle Button)** - **Show Port Labels** shows/hides labels for the switch's buttons. **1 Toggle Button**, when enabled, displays only the first button (i.e. the first input port) of the switch (see Tip below).
- **Size (Small, Medium, Big)** - determines the size of a switch whose style is set to Buttons.
- **Skin Bitmap** - see below, **Panel Control Skins**.

---



**Note:** If you are using a switch to toggle between two states (e.g. on/off, engage/bypass, etc.), you can use the **1 Toggle Button** option to display a single switch button (On, Bypass, etc.) instead of two buttons.

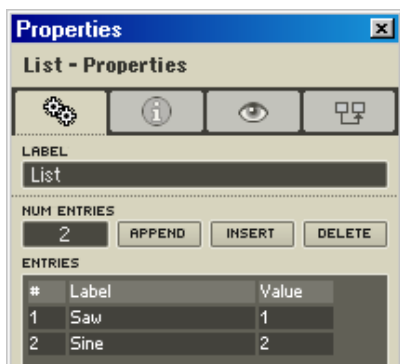
---

## List



### Different types of lists

A list is a panel control whose setting (selected option) determines the value that its source module (  List  List) outputs to other modules in the structure. You define its options and their corresponding values in the Entries list box (Properties dialog, Function page):



Properties dialog of a list (Function page)

**Num Entries** - specifies the number of entries (options) in the list.

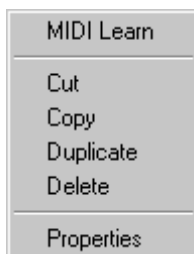
- **Append, Insert, Delete** - **Append** appends a new entry to the end of the list, **Insert** inserts a new entry after the selected entry, and **Delete** deletes the selected entry.
- **Entries (#, Label, Value)** - **#** displays the entry number, **Label** specifies the entry label (i.e. the text that will appear in the list panel control), **Value** specifies the entry value.

Along with your mouse, you can use MIDI to change switch settings (see below, MIDI Control).

You can change a list's panel appearance in its Properties dialog (Appearance page), just as you would change a switch's panel appearance (see above, **Switch**.)

## Context Menu

The following context menu options appear when you XP Right-click /OS X: Ctrl.+click) on any of the above panel controls (fader, knob, button, switch, list):



Context menu of a fader, knob, button, switch and list

- **MIDI Learn:** Enables MIDI Learn for the control, which helps you assign an external MIDI control (e.g. a knob on a MIDI keyboard) to a panel control. (For details, see above, **REAKTOR Toolbars, Ensemble Toolbar.**)
- **Set to Default:** Sets the control to its default value (as specified in **Properties** dialog, Function page).
- **Show in Structure:** Opens the structure in which the panel control's source module is located.
- **Properties** Opens the control's Properties dialog.

## 13.4. Panel Control Skins

REAKTOR enables you to customize the appearance of several panel controls by applying skins to them: faders, knobs, buttons, lists, switches, Receive modules, lamps, and meters.

### Fader Skins

There are two types of fader skins: single-picture skins and animation (multiple-picture) skins.

In a single-picture skin, the picture is used as the handle (not the body) of the fader. If the picture is resizable horizontally or vertically (Picture **Properties** dialog), it is resized to the horizontal or vertical size of the original REAKTOR fader handle. If not, the handle size is the same as the picture size.

You can use the Head (Pixels) property (**Properties** dialog, Appearance) to add a head to your custom fader handle (see below). Setting Head (Pixels) to 0 places the handle exactly within the fader groove (no head). Setting Head to N (1, 2, 3, etc.) creates an N-pixel-wide handle head.

**Picture Properties**

NAME  
Fader-Pad\_Behringer

TRANSPARENCY  
☒ Has Alpha Channel

NUM ANIMATIONS  
1

☐ Horizontal

ANIMATION HEIGHT  
50

ANIMATION WIDTH  
25

RESIZABILITY  
☐ Vertical ☐ Horizontal

BORDER TOP  
0

BORDER LEFT  
0

BORDER BOTTOM  
0

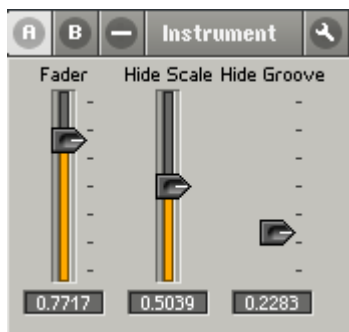
BORDER RIGHT  
0

PREVIEW

OK CANCEL

Picture Properties dialog dialog of a single picture skin bitmap for a fader

In an animation skin, the entire fader (not just the handle) is replaced by the picture; thus the fader size is determined by the picture size, and the Resizability (Picture Properties dialog) and Length (Properties dialog, Appearance page) settings are ignored. The number of fader states is equal to the number of animation frames in the picture.



Faders with hidden scale and hidden groove

For all fader skin modes – single-picture, animation, and none – the Hide Scale and Hide Groove options (Properties dialog) hide the fader scale graphics (tickmarks) and groove.



Properties dialog (Appearance page) of a fader

## Knob Skins

A knob skin is always treated as an animation. The entire knob is replaced by the animation picture; thus the fader size is determined by the picture size, and the Resizability (Picture Properties dialog) and Length (Properties dialog) settings are ignored. The number of knob states is equal to the number of animation frames in the picture.



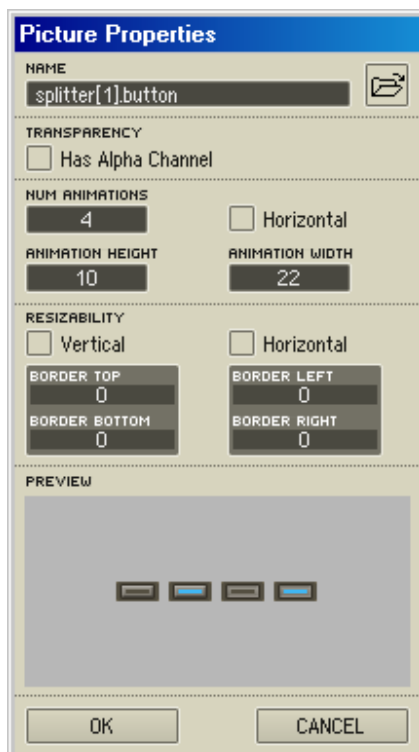
Picture Properties dialog of a skin bitmap for a knob with its animation



Knobs with different skins

### Button (List, Switch, Receive) Skins

A button skin uses a four-frame animation picture to define its four states (in this order): Off state Up, On state Up, Off state Down, On state Down. If the picture is resizable horizontally or vertically (Picture Properties dialog), it is resized to the horizontal or vertical size of the original button. If not, the button size is the same as the picture size.



Picture Properties dialog of a skin bitmap for a button with its four animation states  
The List, Switch, and Receive modules can all have button-type skins if their style is set to Buttons (Properties dialog, Appearance page).



A list module and a switch module which use skin bitmaps for their button states

## Lamp Skins

A lamp skin uses a two-frame animation picture to define its two states (in this order): Off state, On state. If the picture is resizable horizontally or vertically (Picture Properties dialog), it is resized to the horizontal or vertical size of the original lamp (as determined by Size X and Size Y in Properties). If not, the lamp size is the same as the picture size.

## Meter Skins

There are two types of meter skins: on/off and animation.

An on/off skin uses a two-frame animation picture to define its two states (in this order): Off state, On state. If the picture is resizable horizontally or vertically (Picture Properties dialog), it is resized to the horizontal or vertical size of the original meter (as determined by Size X Segment and Size Y Segment in Properties). If not, the meter size is the same as the picture size.

An animation skin uses a multiple-frame animation picture to define its states. If the picture is resizable horizontally or vertically (Picture Properties dialog), it is resized to the horizontal or vertical size of the original meter (as determined by Size X Segment and Size Y Segment in Properties). If not, the meter size is the same as the picture size. The number of meter states is equal to the number of animation frames in the picture. The number of segments is determined by the number of animations frames; thus Number of Segments (Properties dialog) is ignored

## 13.5. Connection Properties of Panel Controls

Most panel controls have a Connection page (indicated by the MIDI-connector icon) with the following sections and settings:

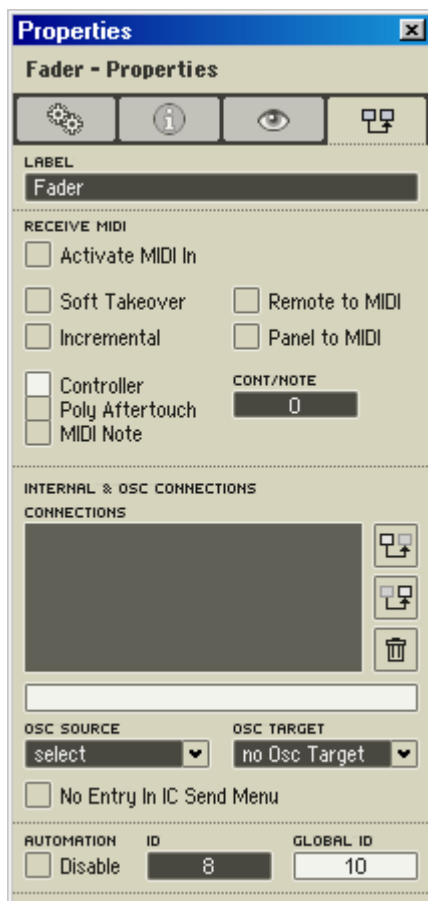
### Receive MIDI

**Activate MIDI In:** When activated, the panel control values can be changed by incoming MIDI events. You can choose between MIDI Controller and Polyphonic Aftertouch messages, and you can specify the controller or Aftertouch-note number.

### MIDI

**Soft Takeover:** When enabled, the control will not be affected until the incoming value passes the control's current value (either going up or going down). This option prevents sudden jumps in the control value when the position of the software controller does not match that of the hardware controller, which can happen, for example, after an onscreen change or a snapshot recall.





Connection page in the Properties window of a panel control (here: fader)

**Incremental:** If active, incoming MIDI messages will be interpreted as coming from an incremental controller. Incremental controllers (often called endless rotaries or continuous controllers) are found on many MIDI control surfaces, including Native instruments' 4Control.

**Panel to MIDI:** If enabled, REAKTOR will send MIDI events whenever the panel control is changed with the mouse.



**Remote to MIDI:** If enabled, causes MIDI events to be output by REAKTOR when the control is changed by incoming MIDI events (cf., Activate MIDI In). When working with a sequencer, bear in mind that a feedback loop may result if the sequencer both sends MIDI data to REAKTOR and receives MIDI from it.

**Controller, Poly Aftertouch, and MIDI Note:** These determine whether the panel control receives and/or sends MIDI Controller, Polyphonic Aftertouch (Key Pressure), or MIDI note messages.

**Cont/Note:** Sets the number of the MIDI controller or note that is assigned to the panel control.

## Connection

The Connections section on the Connection page is available for **Fader, Knob, Button, Switch, XY, Lamp, Meter, Multi Picture** and **Multi Text** modules. Note that it is also available for all MIDI In and MIDI Out modules, thereby allowing wireless communication between different instruments and macros. This section controls internal, wireless communication of data within REAKTOR as well as enables OSC connections between REAKTOR applications running on different computers linked by OSC. Two operations are required to create an internal connection:


- Select the panel control you want to use as the control master and press the  topmost button to the right of the Connections list.
- Select the panel control you want to use as the control slave and press the  middle button to the right of the Connections list.

You can make those selections in any order and one master can control several slaves, in which case the Connection list for the master becomes longer.

To make OSC connections use the two drop-down menus labeled **OSC Source** and **OSC Target**. The **OSC Source** drop-down menu displays controls on other REAKTOR computers, from which values have already been received over OSC. (If REAKTOR has not received any OSC data, the drop-down list will be empty.)

The **OSC Target** drop-down menu shows other OSC REAKTOR computers. This is the same list as found in the **OSC Settings...** window of the REAKTOR **System** menu. Use the **OSC Target** drop-down menu to tell the OSC REAKTOR target computer to place this control in its **OSC Source** drop-down menu.

Any existing internal or OSC connection for a module will be listed as an entry in the Connections list. If the module is master of a connection, the entry will have the prefix “to”, whereas if it is the slave, it will have the prefix “from”.

To delete an entry, select it and click the  **Delete** button to the right of the Connections list.

The two settings at the bottom of the Connection page determine whether the panel control should appear as a selectable parameter in the plug-in hosts parameter automation list (**Disable Automation**) and what position it should have in the list (**ID**). If you enter a number in the ID field that is used by another control in your ensemble, the ID will be swapped with the other control.

Make sure that the number entered in the **Max Automation ID** field in the instrument properties is high enough to ensure that this parameter can be shown in the plug-in host's parameter automation list.

Two-dimension panel controls like **XY** and **Multi Picture** actually have two automation IDs. The second ID will automatically be set to one greater than the first ID and cannot be edited. This ensures that these two parameters appear consecutively in your host software's parameter automation list.

## 13.6. Editing the Panels

Just like the modules in a structure, the controls in the panel can be edited with **Duplicate** and **Delete**. However, remember that these operations always have a direct effect on the respective structure. For example, if you delete a control from the panel you are also deleting the corresponding source module in the structure, because the two are inseparable. We recommend that you carry out these operations only in the structure so that you can keep an eye on the consequences of your actions.

**Moving** controls in the panel, on the other hand, has no effect on the structure. Simply click the left mouse button on the label of the control you want to move and drag it with held mouse button to the desired position. Once all controls

have been arranged, it is best to activate the  **Panel Lock** function. When **Panel Lock** is enabled, panel elements can no longer be moved around. The **Panel Lock** function is set using the context menu of the panel window or simply by clicking on the wrench icon in the Instrument header (a screw icon will appear indicating that the panel is locked down).

# 14. Panel Operation

## 14.1. Mouse Control

### Fader



To change the fader setting, drag its handle to the desired position.

### Knob

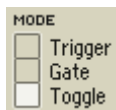


To change the knob setting, drag the mouse up and down over it.

### Button



You can set a button's operating mode (**Trigger**, **Gate**, or **Toggle**) and its **On Value** and **Off Value** in its **Properties** dialog (Function page).



- **Trigger:** Pressing the button generates an event with the specified **On Value**. Releasing the button does not generate an event.
- **Gate:** Pressing the button generates an event with the **On Value**. Releasing the button generates an event with the **Off Value**.

- **Toggle:** The button has two states; hence the term “toggle”. Pressing it once switches it on and generates an event with the specified **On Value**. Pressing it again switches it off and generates an event with the specified **Off Value**.

When connecting a button to an audio input port, **Trigger** mode behaves the same as **Gate** mode (i.e. the signal returns to its **Off Value** when the button is released).

## List



To change the list setting, click on an item to select it.

## Switch



Pressing one of the buttons in a switch lets the corresponding input signal pass through the switch, while blocking all other input signals. (To see this, you need to view the switch in its Structure window.) Only one switch button can be active at a time, and therefore only one input signal can pass through at a time.

## Drop-Down Menu (Switch and List modules)



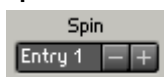
In a drop-down menu, you select one item from a group of items. Click (and release) on the menu to display all its items, then click on an item to select it. Switch and List modules can be displayed as drop-down menus (**Properties** dialog, Appearance page).

## Text Panel (Switch and List modules)



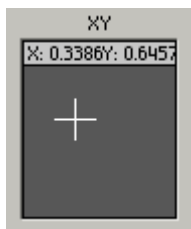
In a text panel, you select one item from a group of items. If there are more items than can fit in the text panel, a scrollbar is displayed. Click on an item to select it. Switch and List modules can be displayed as text panels (**Properties** dialog, Appearance page).

## Spin (Switch and List modules)



In a spin control, you select one item from a group of items. To select an item, click the +/- buttons, or click the spin text box and drag your mouse up or down. Switch and List modules can be displayed as spin controls (**Properties** dialog, Appearance page).

## XY



XY controls two parameters at once. Click within the XY field (as set by Size X and Size Y in the Properties dialog) and drag up/down to control the Y parameter, or left/right to control the X parameter.

# Custom Controls



Splitter is an instrument from the REAKTOR Library

It is possible to create sophisticated custom panel controls in REAKTOR: knobs and faders with beautiful handmade skins, great-looking XY modules that send values to many input ports simultaneously, internal-connection or sequencer driven automation that changes knob/fader settings in real-time. The design and behaviour of custom controls is defined by the instrument creator. Please refer to a specific ensemble's documentation (or, better yet, reverse-engineer the ensemble) to learn more about its custom controls.



GoBox is an Ensemble included in the library

## 14.2. Using Keys to Change Control Settings

Faders, knobs, and switches can be controlled with keys on a computer keyboard. Note that you must select (click on) the fader, knob, or switch to enable the keys to work.

The **↑/↓** and **PgUp/PgDn** keys change fader and knob positions:

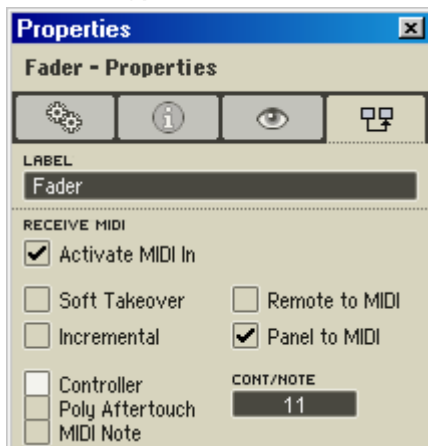
Key: Value Change:

- **↑** +Step
- **↓** -Step
- **PgUp** +10 × Step
- **PgDn** -10 × Step

The **↑/↓** keys toggle through a switch's options (buttons, menu items, etc.).

## 14.3. MIDI Control

### MIDI Data Types



*Properties dialog of a fader, Connection page*

If **Activate MIDI In** is enabled in the **Properties** dialog (Connection page) of a panel control, the control can be operated via MIDI.

- **Controller:** When enabled, MIDI Controller messages (received from an external MIDI device, or internally from within REAKTOR) operate the panel control.



- **Poly Aftertouch:** When enabled, MIDI Poly Aftertouch messages (from an external MIDI device, or from REAKTOR) operate the panel control.
- **MIDI Note:** When enabled, MIDI note messages (from an external MIDI device, or from REAKTOR) operate the panel control. The velocity determines the control position.
- **Cont/Note:** The number of the MIDI controller (if Controller is enabled) or MIDI note (if **Poly Aftertouch** or **MIDI Note** is enabled) whose MIDI messages are used to operate the panel control. You can set the Cont/Note value manually, for **Cont** you can use **MIDI Learn** (see below) to set it automatically.

Fader and knob controls change their positions according to the received MIDI messages.

When using MIDI to operate a button control, you'll find that it turns on only when the received MIDI Controller or Poly Aftertouch message has a value greater than 63. You can also use Note On/Off messages to operate buttons.

When using MIDI Controller or Poly Aftertouch messages to operate a switch or list control, the switch/list selects the option that corresponds to the received MIDI value. The range of possible values (0 to 127) is divided into equal-sized regions according to the number of switch/list options. For example, a switch with four inputs (i.e. four options) would have the following four regions: 0-31, 32-63, 64-95, and 96-127. Note that 0 always selects the lowest switch/list option, and 127 always selects the highest.

## MIDI Learn

The **MIDI Learn** function is switched on with the corresponding button on the toolbar. It is identified by an icon representing a MIDI socket and the letter **L**. It is a very efficient tool for assigning MIDI messages to control elements on the panel.



Select the control element which is to be MIDI-controlled, click on the **MIDI Learn** button, and send the MIDI data that you want to use for controlling (by moving the hardware wheel, knob, fader, pedal or other controller). To MIDI-fy other controls just repeat this operation.

REAKTOR automatically detects whether the controller data comes from a standard MIDI controller or from an incremental controller. The panel element's **Incremental** mode switch is set accordingly. In the rare case that the **MIDI Learn** function chooses the wrong mode, simply repeat the operation or set the correct mode in the Properties of the panel element manually.

## Incremental

You need to enable this entry in the **Properties dialog** of controls or MIDI source modules if you want to control them using a MIDI Controller that sends incremental values.

## Soft Takeover

When a fader or knob is operated by MIDI remote, it normally jumps straight to the received controller value. Such a jump can be quite noticeable in the sound, depending on the controlled parameter (e.g., amplifier level) and is often undesirable. Such jumps can be avoided by enabling **Soft Takeover** in the Properties dialog of the relevant controls. The control will only move when the value received via MIDI reaches or goes past the current position.

## 14.4. MIDI Out

When **Panel to MIDI** is enabled in the Properties dialog of a panel control, all control changes (knob movements, button clicks, etc.) are translated to MIDI messages and output by REAKTOR to the MIDI output port(s) specified in the Audio Setup dialog, MIDI page.

When **Remote to MIDI** is enabled, the MIDI events that are received by the control (when **Activate MIDI In** is on) are also sent to the MIDI output port(s). Take care, however, when connecting to a sequencer that sends MIDI messages to REAKTOR and also receives MIDI messages from REAKTOR. In such cases, a feedback loop may result.

## 14.5. Customized Panels

It is possible to create fully customized panels in REAKTOR. You can design background bitmaps, displays that react to user input, even your own custom controls. (See above, **Custom Controls**.) You can add rectangular bitmaps to the panel, or use alpha-channel transparency to add arbitrarily shaped bitmaps. You can use the **Snap Value** module to save the current setting of a custom control with a snapshot. And so on.

### Customized fader

REAKTOR allows you to use 24-bit Bitmap (\*.bmp) and 32-bit uncompressed Targa (\*.tga) images in many places: as instrument and ensemble panel backgrounds, as instrument and primary macro structure icons, as panel control (e.g. knob, fader, etc.) skins, and in **Multi Display**, **Poly Display**, **Picture**, and **Multi Picture** modules.

The advantage of using a Targa image is that it enables you to add transparency to the image (see below, Transparency).

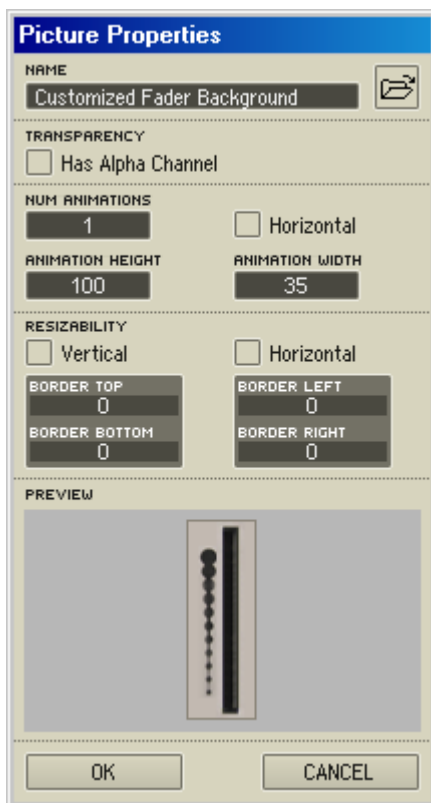
## Picture Properties Dialog



### *Properties dialog (Appearance page) of a fader*

You can load a picture using the drop-down **Select Picture** (or **Object Picture**, **Background Picture**, **Structure Icon**) menu in the Properties dialog (Appearance page) of any REAKTOR object that can display pictures: instrument, macro, Picture and Multi Picture modules, etc. To load a picture file from disk, select **Open from File...**; to load a picture from memory, select it by name from the lower half of the menu.

Opening a picture file from disk opens the **Picture Properties** dialog. It is in this dialog box that you make all relevant picture settings.



Picture Properties

---

**Note:** All of the pictures that are loaded in an ensemble are available to all the ensemble objects (in their drop-down picture menus) that can display pictures. Using the same picture more than once does not appreciably increase an ensemble's memory needs. So, as long as you work with a relatively small number of picture files, you can use them as much as you want without dire RAM consequences.

---

Once a picture has been loaded into an object, you can open its Picture Properties dialog by selecting **Picture Properties...** from the object's drop-down picture menu (as discussed above). Bear in mind that any changes you make to the picture will apply to all instances of the picture in the ensemble.

The **Picture Properties** dialog has five sections: **Name**, **Transparency**, **Animation**, **Resizeability**, and **Preview**. Let's look at these, one by one.

## Name

**Name** renames the picture for internal use within REAKTOR. (**Name** doesn't change the name of the picture file stored on your hard drive.)

## Transparency

**Has Alpha Channel** enables/disables a transparency mask using the picture's alpha channel. Alpha channels are available for Targa (\*.tga) images, but not for Bitmaps (\*.bmp). If a Targa image has an alpha channel mask (a task for the creator of the image), enabling **Has Alpha Channel** makes the unmasked portion of the image transparent (invisible). You can use this to display round custom knobs, panel overlays with blank spaces for REAKTOR panel controls, or any other irregular (non-rectangular) image.

## Animation

The Animation part of the Picture Properties dialog enables you to break a single multipart image into a number of separate sub-images (frames). Each frame can be displayed by its index number: 0, 1, 2, ..., N-1 (where N is the total number of frames in the animation).

Animation frames are typically equal-sized vertical “slices” of a single very tall image; e.g. 128 instances of a knob image stacked one beside each other and on top of the other, with each successive instance (from left to right and top to bottom) showing the handle a bit closer to the maximum state position of the knob. Display all these frames in order, and you've got an animation of a knob whose handle is moving from its minimum to maximum position.

You can also use horizontal frames, by enabling the **Horizontal** option, but we recommend sticking with vertical frames, because REAKTOR must work a bit harder to process horizontal frames.

- **Num Animations:** Sets the number of animation frames you want to break the picture into.
- **Animation Height:** In vertical mode (i.e. when **Horizontal** is disabled), **Animation Height** sets the height (pixels) of each frame.
- **Animation Width:** In horizontal mode (i.e. when **Horizontal** is enabled), **Animation Width** sets the width (pixels) of each frame.
- **Horizontal:** Toggles between horizontal mode (**Horizontal** enabled) and vertical mode (**Horizontal** disabled).



Picture Properties dialog of a skin bitmap for a knob with its animation

In vertical mode, the **Num Animations** and **Animation Height** values are interdependent. When you set one, the other automatically sets itself accordingly. For example, if your picture is 4000 pixels tall, and you set **Num Animations** to 40, **Animation Height** automatically sets itself to 100 (pixels):  $4000 / 40$ . Or, if you set Animation Height to 50 (pixels), Num Animations sets itself to 80:  $4000 / 50$ .

In horizontal mode, the Num Animations and Animation Width values are similarly interdependent.

## Resizability

The **Resizability** part of the Picture Properties dialog serves two main purposes. It reduces the size of ensemble files (\*.ens) by enabling small pictures to be tiled so that they can fill large panel areas (i.e. instrument panel backgrounds). And it uses scaling to enable pictures (e.g. fader skins) to fit their associated objects (e.g. panel controls).

- **Vertical:** enables vertical tiling and scaling.
- **Horizontal:** enables horizontal tiling and scaling.
- **Border Top, Border Bottom, Border Left, Border Right:** cause a tiled picture to overlap itself in places. You can, for example, create a panel control of which a part does not change in form or color on its X or Y axis (see the example of the knob control below, which uses the **Resizability** option while Horizontal is activated).



Left: Original picture. Right: Picture enlarged using the Resizability. The text was added using the Text module.

## Preview

Preview displays a preview of the picture with the current Picture Properties dialog settings applied to it.

## 15. Snapshots



The Snapshots window

Snapshots (aka patches, programs, presets) enable you to store and recall an instrument's sounds. When you create a snapshot, the current settings of all the instrument's panel controls (knob/fader positions, list box and switch settings, button states, etc.) and MIDI controllers are stored in the snapshot. When you recall a snapshot, all the instrument's controls are restored to the settings they were in when the snapshot was originally created. Each REAKTOR instrument can store up to 2048 snapshots: 16 banks x 128 snapshots per bank.

---

**Note:** Ensembles can have snapshots too. See below, Linking Snapshots.

---



## 15.1. Control ID Numbers

Every REAKTOR panel control has a unique ID number, as displayed in the **ID For Snapshot Files** field of the control's Properties window (Function page).



ID For Snapshot Files field in an Properties window, Function page

---

**Note:** Do not change these control ID numbers!

---

REAKTOR lets you change them. But this is not generally a very useful (or smart) thing to do. Snapshots assign specific values to specific control ID numbers. For example, Snapshot1 might assign the value .5 to a knob with the ID number 21, the value .75 to a fader with ID 22, and so on. If you change the knob and fader ID numbers, it will change the values that Snapshot1 assigns to them. Thus your beautifully sculpted snapshots will all be broken.

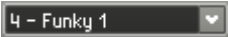

---

**Note:** If you load a snapshot file into a different instrument than it was created for, the snapshots will assign quasi-random values to the new instrument's controls (provided that the two instruments use the same approximate range of control ID numbers). Experimentalists might use this to coax new sounds out of old instruments.

---

## 15.2. Recalling Snapshots

There are three different methods you can use to recall an instrument's snapshots:

- You can use your mouse to select snapshots from the **Snapshots** drop-down menu in the instrument panel header. If you click on the menu, you can use your computer keyboard's Up/Down arrow keys to select previous/next snapshots. 
- You can select snapshots from the **Snapshots** window (**View->Show Snapshots**, or **F6** or  **Snapshot Button** in the Ensemble panel toolbar). If you do, make sure that the desired instrument is selected in the **Select Instrument** list box at the top of the Snapshots window.
- You can select snapshots by issuing MIDI Program Change messages from a MIDI keyboard (or other MIDI controller). For this to work, the **Recall by MIDI** option must be enabled in the instrument's Properties

dialog (**Function Page**). The MIDI Program Change message selects a snapshot by its number (1-128): MIDI Program Change 0 selects snapshot number 1, MIDI Program Change 1 selects snapshot 2, and so on.

---

**Note:** REAKTOR builders, you can use the Snapshot module to recall, store, randomize, and morph snapshots.

---

## 15.3. Linking Snapshots


By default, snapshots are stored and recalled independently for each instrument. For example, say you have an ensemble that contains two instruments, Inst1 and Inst2. Selecting a new Inst1 snapshot does not select a new Inst2 snapshot, and vice versa.

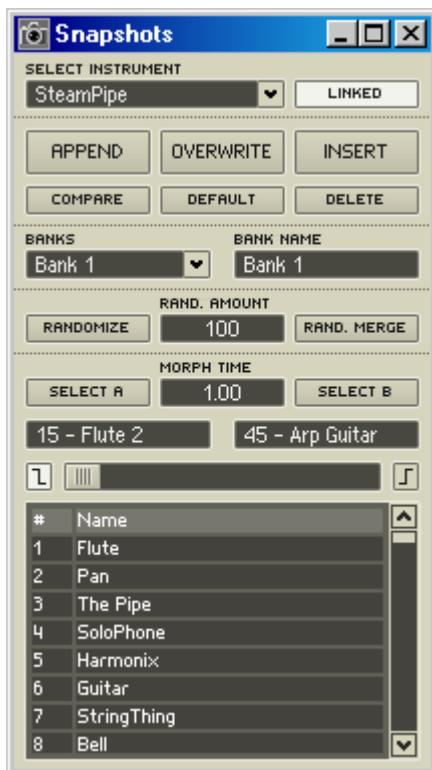
Sometimes this is just what you want. Other times, you might want to select snapshots for multiple instruments at the same time. You can do this by linking snapshots:

1. Use the Panelset bar to display the ensemble panel and all instrument panels in the Ensemble Panel window.
2. Enable each instrument's **Recall by Parent** option (**Properties** dialog, **Function page**).
3. Select an existing ensemble snapshot, or create a new one and select it.
4. Select the desired snapshot for each instrument.
5. Save the ensemble snapshot. (For help, see below, Managing Snapshots.)  
From now on, when you select this ensemble snapshot, it will cause all the instruments to select the snapshots you specified for them in step 4.

## 15.4. Managing Snapshots

You can manage (create, save, delete, etc.) snapshots in the Snapshots window. To open this window:

- Select **View->Show Snapshots** from the main menu.
- Or press **F6**.
- Or XP: Right-click /OS X: Ctrl.+click) on an instrument panel header and select **Snapshots** from the context menu.
- Press the  **Snapshot Button** in the Ensemble panel toolbar.



The Snapshots window

In the **Select Instrument** drop-down menu, you select the instrument whose snapshots you want to manage. You can also select the ensemble to manage ensemble snapshots.

If the **Linked** option is enabled, the Snapshots window automatically shows the snapshots of the instrument that is currently selected in the Ensemble Panel window. If **Linked** is off, you must use the **Select Instrument** menu to manually choose which instrument's snapshots to display. Most experienced users work with **Linked** on.

Below the **Select Instrument** menu is a set of six buttons for managing snapshots:

- **Append** saves the current instrument/ensemble settings as a snapshot to the first empty slot in the snapshots list. If the current snapshot bank is full, **Append** saves the snapshot to the first empty slot in the next

bank. If there are no more empty snapshot slots, **Append** does nothing. (Each instrument/ensemble can have a total of 2048 snapshots: 16 banks x 128 snapshots per bank.)

- **Overwrite** replaces the selected snapshot with the current instrument/ensemble settings. Note that when you overwrite a snapshot, you lose its original settings.
- **Insert** inserts the current instrument/ensemble settings as a new snapshot directly after the selected snapshot. Note that this can cause snapshots to move from the current bank to the next bank.

Important: You need to click the **Append**, **Overwrite**, and **Insert** buttons *twice* for them to work correctly. The first click lights the button and places a blinking cursor in the appended, overwritten, or inserted snapshot, giving you the opportunity to type a name for it. The second click un-lights the button and saves the appended, overwritten, or inserted snapshot. Remember to click twice! If you forget the second click, you might end up doing something very different than you intended.

- **Compare** compares the current instrument/ensemble settings with the original settings of the selected snapshot. (See Comparing, below.)
- **Default** changes the current instrument/ensemble settings to their default values (as specified in each instrument/ensemble control's Properties dialog, Function page). Note that clicking **Default** changes the selected snapshot's settings, but does not save the changed snapshot. To do this, you must use **Overwrite**.
- **Delete** deletes the selected snapshot(s). Note that **Delete** creates holes (empty slots) in the snapshots list; you can use the Banks menu's Sort command to remove these holes. (See below, Banks Menu.)

## 15.5. Renaming and Copying Snapshots

To rename an existing snapshot in the Snapshots window:

- Double-click on the snapshot, type the desired name, and click on the **Enter** key to save the renamed snapshot.
- Or select the snapshot, click the **Overwrite** button, type the desired name, and then click **Overwrite** a second time to save the renamed snapshot.

To copy an existing snapshot in the Snapshots window:

- Select the snapshot, click the **Append** button, rename the appended snapshot if desired, and then click **Append** a second time to save the appended snapshot. Note that this copies the snapshot to the first empty slot in the snapshots list.

- Or select the snapshot, click the **Insert** button, rename the inserted snapshot if desired, and then click **Insert** a second time to save the inserted snapshot. Note that this copies the snapshot to a new slot below the originally selected slot in the snapshots list.

## 15.6. Comparing Snapshots

The **Compare** button is used for two main tasks:

- To compare a snapshot with a modified version of the same snapshot. Listening back and forth between the original and modified versions can help you create better snapshots.
- To compare two different snapshots.

The theory behind Compare is simple. The modified (or different) snapshot is stored in the Compare buffer, and the Compare button is used to toggle between the original snapshot and the modified (or different) snapshot.

To compare a snapshot with a modified version of the same snapshot:

1. Select a snapshot in the Snapshots window.
2. Make sure the **Compare** button is off (unlit).
3. Modify the snapshot control settings as desired.
4. Click twice on the **Compare** button; the first click turns it on (lit), the second off (unlit). The modified snapshot is now stored in the Compare buffer.
5. Use the **Compare** button to toggle between the original and modified snapshot versions.
6. Repeat steps 2-5 for further modifications.

To compare two different snapshots:

1. Select a snapshot in the Snapshots window.
2. Select another snapshot. The first snapshot is now stored in the Compare buffer.
3. Use the **Compare** button to toggle between the two snapshots.

---

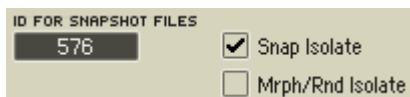
**Note:** If you are modifying a snapshot and accidentally select a different snapshot, you can recover your modifications by clicking on Compare right away (before you make any changes to the new snapshot)

---

## 15.7. Snap Isolate

When you select a snapshot, the panel controls and MIDI controllers jump to their positions as stored in the snapshot. In some cases, you might not want this to happen. For example, you might create a sequencer with a BPM (tempo) knob whose settings you want to be snapshot-independent.

To prevent a panel control or MIDI controller from jumping to its snapshot-designated position, enable the **Snap Isolate** option in its Properties dialog (Function page).



Snap Isolate in an Properties window, Function page

## 15.8. Banks Menu



Banks menu in the Snapshots window

The Banks menu is divided into two parts. The upper part is used for selecting banks (of snapshots). The lower part is used for managing (creating, sorting, cloning, etc.) banks.

- To select an existing bank, click on its name in the upper part of the Banks menu. The new bank's snapshots will be displayed in the snapshots list.

- To rename an existing bank, select it, and type the desired name in the Bank Name field (to the right of the Banks menu).

The lower part of the Banks menu offers these commands:

- **New** creates a new, empty bank and saves it to the first free slot in the banks list. For example, if an instrument had a Bank 1 and a Bank 3, **New** would create a Bank 2; if it had a Bank 1, 2, and 3, **New** would create a Bank 4; and so on. Each instrument/ensemble can have up to 16 banks (with up to 128 snapshots per bank).
- **Sort** sorts the selected bank's snapshots by number, and removes all <empty> slots (holes) from the snapshots list.
- **Init** initializes all snapshot. This erases all snapshots!
- **Clone** creates a copy of the selected bank and saves it to the first free slot in the banks list.
- **Save** saves the selected bank's snapshots to a snapshot file (\*.ssf).
- **Load** loads the snapshots from a snapshot file (\*.ssf) into the selected bank. You will be given the option to write these new snapshots over the bank's current snapshots (thus deleting the current snapshots), or to append the new snapshots to the end of the bank's snapshots.
- **Delete** deletes the selected bank.

---

**Note:** If you delete a bank by mistake, don't panic! Simply use REAKTOR's **Undo** command to un-delete it.

---

## 15.9. Randomizing Snapshots

The Snapshots window provides several controls that you can use to add a degree of randomization to your snapshots:



Randomization row in Snapshots window

- Clicking on the **Randomize** button randomizes all of the selected instrument's panel controls, except those whose **Random Isolate** option is enabled (Properties dialog, Function page). (See below, Tip.)
- The value in the **Rand. Amount** field (0-100, which corresponds to 0%-100%) determines the maximum amount of randomization that the **Randomize** button can deliver. Clicking on **Randomize** can change a control's current setting up to **+/- Rand. Amount %** of the control's range.

For example, if a knob with a range of -1 to 1 is set to its middle point (0), and **Rand. Amount** is set to 25 (25%), clicking on **Randomize** can change the knob's value to anywhere from -.5 to .5 (0 +/- (25% of 2)). If the knob is set to -.5, and **Rand. Amount** is set to 50 (50%), clicking on **Randomize** can change the knob's value to anywhere from -1 to .5 (-.5 +/- (50% of 2)). Note that a control can never be randomized to a value beyond its Min/Max range.

- The **Rand. Merge** button works in conjunction with the **Select A** and **Select B** buttons that are used to select snapshots A and B for morphing (see below, Snapshot Morphing). Clicking on **Rand. Merge** spawns a “child” snapshot whose panel-control values are all randomly placed between their values in snapshot A and snapshot B. The degree of randomness is determined by **Rand. Amount**. If, for example, **Rand. Amount** is set to 50 when you click **Rand. Merge**, the child snapshot's control values will all be exactly halfway between their values in snapshot A and snapshot B. If **Rand. Amount** is set to 100, the child snapshot's control values can all end up anywhere between their values in snapshot A and snapshot B. And so on.

---

**Note:** To make a panel control immune to randomization, enable its **Random Isolate** option (Properties dialog, Function page); the **Randomize** button will have no effect on the control. You can use this technique, in conjunction with the **Rand. Amount** value, to limit the amount of randomization an instrument receives.

---

## 15.10. Morphing between Snapshots

The Snapshots window provides a flexible set of controls for morphing between snapshots; i.e. changing an instrument's panel-control settings gradually (over a period of 0-60 seconds) from their values in one snapshot to their values in another snapshot.



The morphing section in the Snapshots window



Here's how you do it:



1. Set your desired morphing time in seconds (0-60) in the **Morph Time** field. This is how long it will take the controls to morph (move) from their current settings to their new settings.
2. Click the **Select A** button to turn it on (lit), and then select your desired snapshot A from the snapshots list.
3. Click the **Select B** button to turn it on, and select snapshot B.
4. Now you're ready to morph: Move the horizontal **Morph** slider to a new position (full left = 100% snapshot A, full right = 100% snapshot B, midway = 50% A and 50% B, and so on). The instrument's controls will move from their current settings to the settings specified by the new **Morph** slider position over the number of seconds specified in the **Morph Time** field.

---

**Note:** Shorter Morph Time values decrease the delay between changing the Morph slider position and having the panel controls complete their morphs. Longer Morph Time values increase this delay.

---

Gradual, incremental change between two states (i.e. two snapshots) is the basis of morphing. Because button and switch settings cannot be changed gradually, REAKTOR does not let you morph them. Therefore, before you begin to morph, you must decide whether to use the button/switch settings from snapshot A or snapshot B. Here's how:

- To use the snapshot A button/switch settings, click the  button to the left of the Morph slider to turn it on (lit).
- To use the snapshot B settings, click the  button to the right of the Morph slider to turn it on (lit).

# 16. Sampling and Resynthesis

In this chapter, you will learn all about REAKTOR's very powerful sampling and resynthesis capabilities.

## 16.1. Sample Management

### Sample Files and RAM

REAKTOR's sampler modules can use mono or stereo .wav or .aif/.aiff sample files of any sample rate and bit depth. If a sample file contains any loop or keyboard-allocation information, REAKTOR finds and processes it.

Before a sampler module can use a sample file, the entire file needs to be loaded into actual – not virtual (see below) – RAM memory. Regardless of the bit depth of the sample file (8-bit, 16-bit, etc.), REAKTOR converts the audio to 32-bit for internal processing. One minute of stereo 32-bit audio at a typical REAKTOR sample rate of 44100 uses 20 MB of RAM. Thus large sample files can use huge amounts of RAM.

Systems that use virtual RAM (the default in Windows, but not in OS X) can allocate a great deal more RAM than is actually available; hence the term “virtual”. In such systems, REAKTOR does not display an error message when you load a sample file which is so large that it requires more actual RAM than is available. Instead, an error message appears later, informing you that the CPU is overloaded.

This error message appears because REAKTOR cannot access the sample audio data on the hard drive fast enough (after first having tried to access it in RAM and not having found it there). Frequently the message is preceded by an unintentional granular sound artifact caused by interruptions of the audio data stream. This situation, particularly undesirable during live performances, can only be avoided by adding more RAM to your system.

### Multiple Use of Identical Samples

If one sample is used by several sampler modules in an ensemble, all the modules access the same sample as it is stored in RAM. This means that you can load the same sample in as many sampler modules as you like without increasing the RAM usage by any appreciable amount. Similarly, the number of polyphonic voices used by a sampler module is irrelevant as far as RAM is concerned.

REAKTOR identifies a sample using the pathname (directory and name) of the sample file from which the sample was loaded. When a new sample is loaded, REAKTOR searches through the list of already loaded sample files. If it finds one with the same pathname, REAKTOR reuses it instead of loading the new sample file.

## Missing Samples

By default, REAKTOR saves the pathnames of the samples that you load into a sampler module, not the actual sample files. If any of these sample files are deleted, renamed, or moved after the ensemble has been saved, they will not be available when the ensemble is reopened.

If this happens, you will get an error message upon opening the ensemble informing you that the sample files are missing. These samples are labeled as **missing** in the File column of the Sample Map Editor. To relocate or replace missing samples, select the sample in the Sample Map Editor, and click on the **Replace** button.

## Storing Samples with Modules

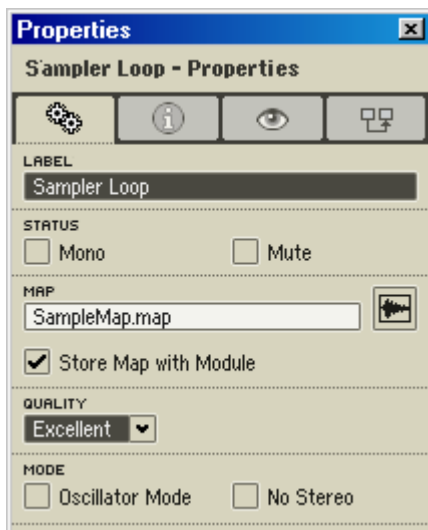
You can avoid the potential problem of missing samples by turning on **Store Map with Module** in the sampler module's Properties dialog (Function page). Doing so saves a copy of all the sample files loaded in the module's sample map with the module (and, by extension, with the ensemble). When you reopen the ensemble, REAKTOR will load the module's sample files directly from the ensemble file, rather than loading the original files from their individual locations on your hard disk.

The advantage of storing samples with a sampler module: you can be sure that these samples will still be there when you open the ensemble, or when another user, on a different computer, opens the ensemble. The disadvantage: depending on the sample files' sizes, you might end up with a very large ensemble file size.

---

**Note:** A common mistake is to save a sampler ensemble without having turned on **Store Map with Module** in the sampler module(s), and then to share this ensemble with other REAKTOR users (e.g. in the REAKTOR User Library). When another user opens the ensemble, the sample files will be missing. Simple fix: Remember to turn on **Store Map with Module** in all sampler modules before saving an ensemble!

---



Properties dialog of a sampler module, Function page

## Sample Analysis

Some sampler modules (e.g. Grain Resynth, Grain Pitch Former, Beat Loop) perform real-time resynthesis of sample files. When you load a sample in a resynthesis module, REAKTOR analyzes the sample file while it is loading. This analysis takes about as long as the duration of the sample. To prevent having to repeat the same analysis every time this file is loaded in a resynthesis module, REAKTOR displays a message dialog asking if you want to save the analysis data into the sample file. Read this message carefully before deciding what to do; it explains the potential danger of embedding analysis data into your sample file. Note that REAKTOR can only save analysis data into a sample file which is not write-protected.

---

**Note:** Once REAKTOR has saved analysis data into a sample file, it assumes that this file has been fully analyzed. This can cause problems. For example, if you modify an analyzed sample file, then load it into a resynthesis module, REAKTOR thinks that the file is still fully analyzed (though it isn't, because you modified it). To get around this, simply rename an analyzed sample file when you modify it. Then, when you load it into a resynthesis module, REAKTOR will re-analyze it.

---

## Sample-Editors

REAKTOR does not have its own internal sample editor. To edit sample files, you have to use an external audio editor (e.g. Windows XP: Sound Forge, WaveLab, Audition, GoldWave etc; OS X: Peak, Spark XL, Audacity etc.). To facilitate this process, REAKTOR enables you to open a sample file in your chosen editor from within the Sample Map Editor window. To do this:

1. Tell REAKTOR where to find your sample editor by entering its pathname in the Preferences dialog: **System** ⇒ **Preferences** ⇒ **Directories: External Sample Editor**.
2. Select the sample file in the Sample Map Editor.
3. Select **Edit** from the drop-down Edit Sample List box to load the sample file in your external sample editor.
4. Edit your sample file as desired, and then save it.
5. Select the sample in the Sample Map Editor, and select **Reload** from the Edit Sample List box to reload the modified sample file.

---

**Note:** Keep in mind that some sample editors will ignore loop information present in sample files. In such cases the loop regions will be lost.

---

## 16.2. Sample Maps

One common use of sample maps is to simulate the sound of acoustic instruments. This is usually done by assigning multiple samples (rather than a single sample) from the instrument to the instrument's pitch range. For example, rather than assigning a single clarinet sample to its entire three-octave range, you might assign a dozen samples to the range: one for the lowest three pitches in the range, another for the next higher three pitches, and so on, all the way to the highest three pitches. The further you stretch a sample (i.e. the more pitches you assign it to), the less natural the simulation will sound. So, to make an instrumental simulation sound natural, you need to use an appropriate number of samples.

Another use of sample maps is to trigger multiple samples with one key. Following are explanations for both uses.

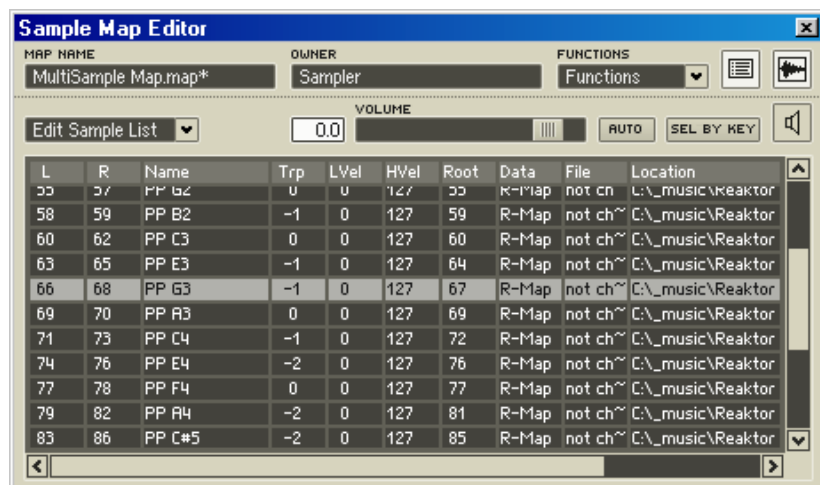
### Multi-Sampling

The repetitive nature of samples makes it difficult to simulate acoustic instruments. With REAKTOR it is possible to vary the sample each time it is triggered.

Usually when a sample is assigned to multiple keys (pitches), the further it is transposed from the root key of the original sample, the less natural it sounds. This happens because the frequency spectrum (overtone) of the transposed sample does not match the spectrum of the acoustic instrument. That is why a human voice transposed an octave higher sounds so unnatural and silly. Samplers try to overcome this limitation by using multi-sampling, wherein each sample is only transposed a small amount from its root key.

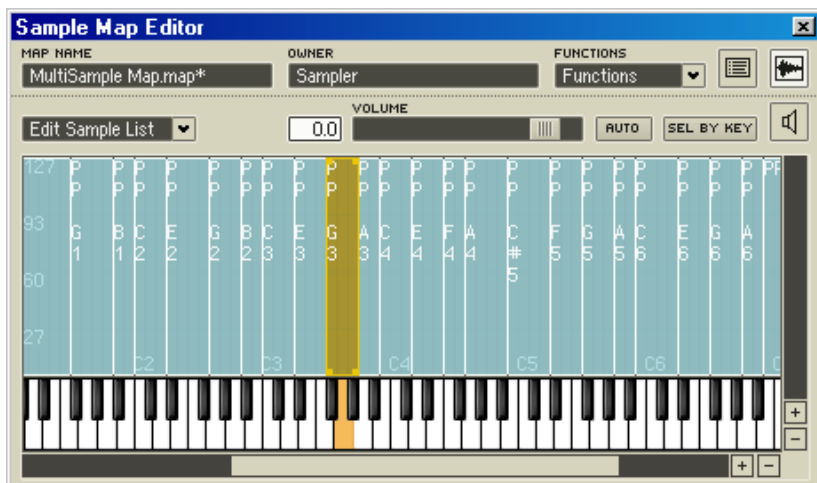
Assuming that a sample map contains multiple samples of the original sound, at least three parameters must be set (in the Sample Map Editor) to control each sample in the map:

- **Root** sets the key (pitch) of the untransposed sample.
- **L** (left split) sets the starting point of the sample key zone.
- **R** (right split) sets the end point of the sample key zone.



Sample Map Editor showing a list of multiple samples

The optimum result is achieved when the Root is kept in the middle of the sample key zone, and the amount of transposition is kept to a minimum.



Part of the Sample Map Editor with a sample map containing multiple samples. The orange key shows the Root key position of the selected sample zone.

Every sampler module has a **P**(itch) input, which is used to select a sample from the map and to control the pitch of the sample. Usually this input is connected to a **Note Pitch** module, which sends the current MIDI note. When a sample is triggered in multi-sample mode, the value received at the **P**-input selects the specified sample and pitch.

The sample module also has a **Sel**(ect) input. If connected, **Sel** overrides the sample selection of the **P** input, which then only sets the pitch. This can result in interesting sound variations.

Sample maps in REAKTOR can also be used to trigger multiple samples with only one key. This is a technique used to further mirror an instrument's dynamics. Velocity is used to achieve this via MIDI. Usually an instrument is sampled playing the same note with different dynamics (e.g., soft and hard). Then the dynamically different samples are triggered by different key velocities, resulting in a more expressive representation of the original sound.

## Drum-Maps

As in Multi-Sampling, a "Drum-Map" uses the **Root Key** and left and right splits to determine the position of the samples in the map. Sometimes the **Root Key** of the original sample is not used or gets overly transposed. With Drum-Maps, it is possible to use any range of keys to a sample (these keys do not have to be anywhere near actual **Root Key** of the drum sample).

As described above, the **Sel**-input is used to override the sample selection of the **P**-input, which then only sets the tuning. The **Sel**-input can be used to achieve interesting sound variations. For example, by connecting the output of a **Gate** module to the **Sel** input, the velocity information is then used to select samples from the map. This way it is possible to trigger many different samples using only one key's velocity information.

## Saving Maps

You can save a Map to disk, separate from the instrument or ensemble that uses it. Under Windows it will be stored with the filename extension **\*.map**. The map file can contain all the sample data that is used by the map, or it can be just a small file with references to the sample files.

The loss of samples can be avoided if the **Store Sample with ensemble** option is activated, which can be accessed via the sampler module's properties window.

## 16.3. Sample Map Editor


You use the **Sample Map Editor** to load, save, edit, map, and loop the sample files that are used in REAKTOR's various sampler modules: Sampler, Grain Resynth, Beat Loop, Sample Lookup, etc. Like the Properties dialog, the Sample Map Editor window can be kept open while you work, and its contents will change to reflect the contents of the currently selected sampler module. If no sampler module is selected, the Sample Map Editor will be empty.

### Opening the Sample Map Editor

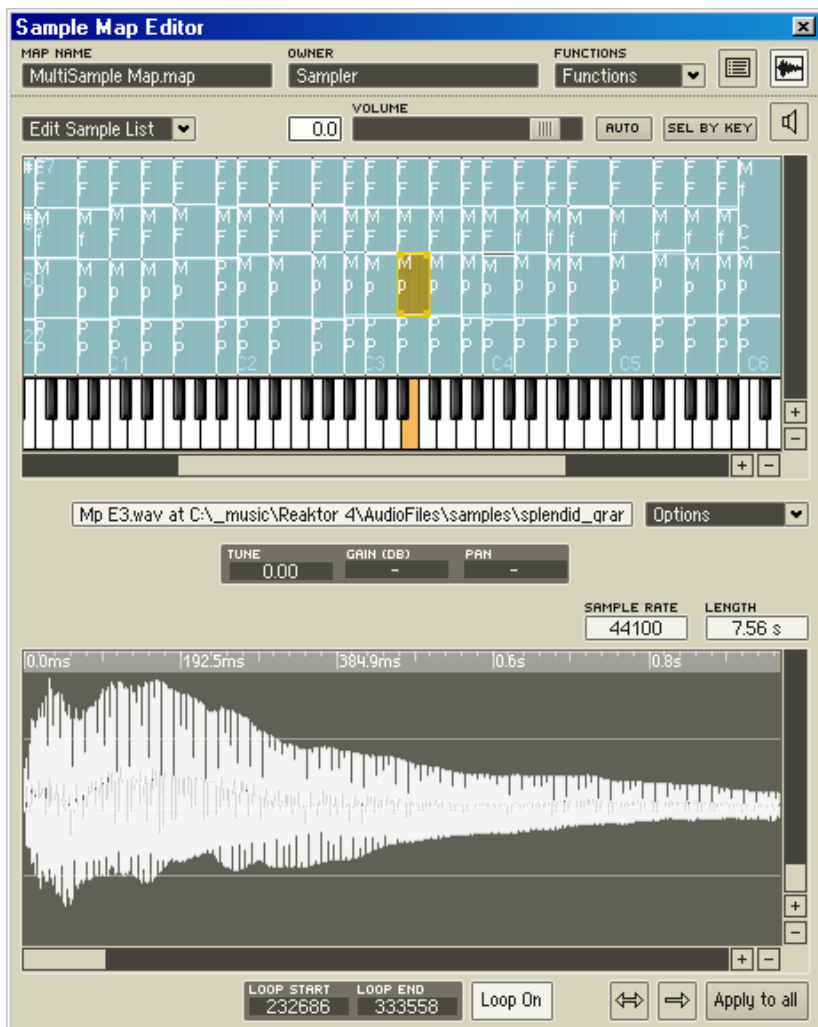
There are four different ways to open the Sample Map Editor. These first two ways open the Sample Map Editor with no sample files loaded in it:

- Select **View->Show Map Editor** from the main menu.
- Or press the **F7** function key.

These next two ways are preferable, since they open the Sample Map Editor with the selected sampler module's samples loaded in it:

- Click on the  **Show Map Editor** button (waveform icon) in the sampler module's Properties dialog (Function page).
- Or Windows XP: Right-click / OS X: Ctrl.-click) on a sampler module's panel display and select **Open Map Editor** from the context menu.



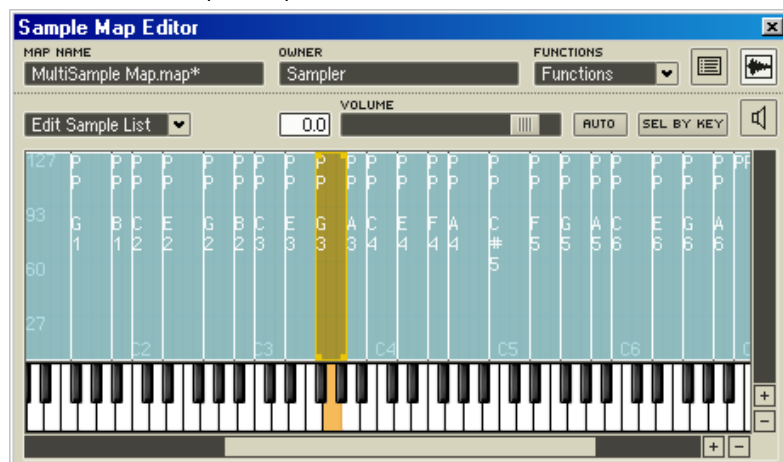


Sample Map Editor with a a sample map containing multiple samples

## Sample Map Editor Components

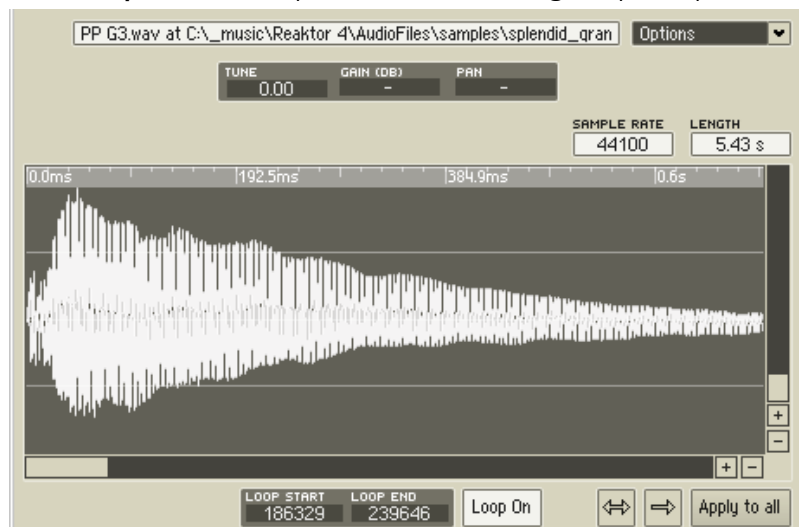
The Sample Map Editor window has two components:

- **Sample Mapper** (upper pane) - used for loading and arranging samples in the sample map.



The Sample Mapper section in the upper pane of the Sample Map Editor window

- **Loop Editor** (lower pane) - used for editing sample loops.




The Loop Editor section in the lower pane of the Sample Map Editor window


Let's take a closer look at both of these components.

## Sample Mapper

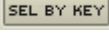
The Sample Mapper contains these controls:


- **Map Name** field: Displays the name of the currently loaded sample map file. If no map file is loaded, “untitled Map” is displayed.
- **Owner** field: Displays the name of the sampler module whose map is displayed.
- **Functions** list box: Contains two commands, Remap to Single Keys and Set transpose to Null for All. For details, see Functions List Box below.

-  **Map View** button (list icon): The Sample Mapper can display samples in list (text) or keyboard (graphic) view. You use the **Map View** button to toggle between these two views. For details, see Map List View and Map Keyboard View below.

-  **Show/Hide Loop Editor** button (waveform icon): Shows and hides the Loop Editor. For details, see Loop Editor below.

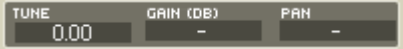
- **Edit Sample List** box: Contains sample-editing commands. For details, see Edit Sample List Box below.

-  **Sel by Key** button: When this option is enabled, incoming MIDI notes will cause the sample selection to change accordingly.

-  **Auditioning** controls: Contains a set of controls for auditioning sample files. For details, see Auditioning Sample Files below.

- **Sample Name** field: Displays the pathname of the currently selected sample file.

- **Options** list box: Contains three commands, Auto-Move RootKey, Ignore RootKey When Loading From File, and Single Key Mode. For details, see Options List Box below.

-  **Tune, Gain, Pan** fields: Display the tuning (in cents), the gain (in dB), and the pan position of the selected sample.

SAMPLE RATE	LENGTH
44100	9.42 s

- The **Sample Rate** and **Length** displays show the sampling rate for the selected sample and its length in milliseconds. These values cannot be edited.

## Edit Sample List Box

The Edit Sample List box contains the following commands:

- **Add:** Adds a sample to the sample map.
- **Replace:** Replaces the selected sample in the map.
- **Save:** Saves the selected sample.
- **Delete:** Deletes the selected sample from the map. Note that this does not delete the sample from your hard disk.
- **Edit:** Opens the selected sample in the external audio editor specified in the Preferences dialog (Directories).
- **Update:** Reloads the currently selected sample. This is typically used to reload a sample file after having edited it.

Note that if no samples are selected, the **Delete**, **Edit**, and **Update** commands are applied to the entire map.

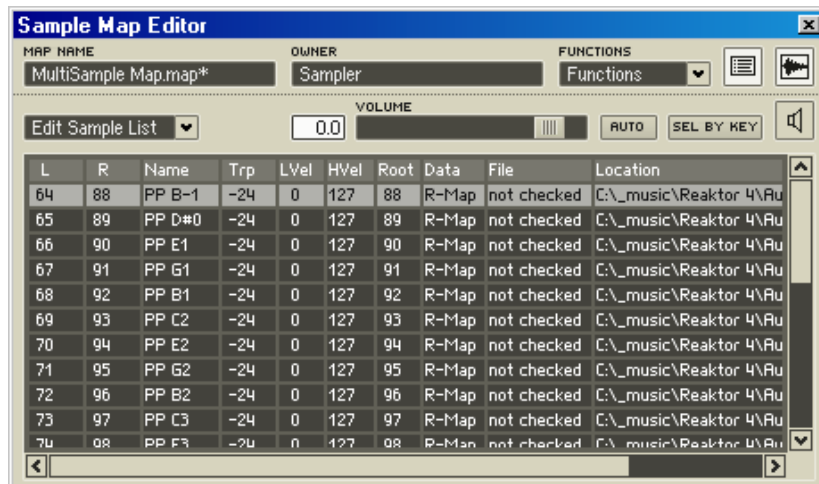
- **Load Map:** Loads a sample map file.
- **Save Map:** Saves the current sample map file. When you save a map file, REAKTOR asks if you would like to save the audio data along with the map. If you say **Yes**, copies of the sample files will be saved with the map; this results in a larger map file, but it also ensures that the samples will all be there if you load the map file in another sampler module. If you say **No**, pathnames of the sample files will be saved with the map; this results in a smaller map file, but it runs the risk of having missing samples if you load the map file in another sampler module.
- **Akai Import:** Opens the Akai Import window for importing sample maps from Akai CDs (S 1000-3000 format). See below for details.

---

**Note:** To use the keyboard to cycle between samples in a map, press Tab (to cycle forward) and Shift + Tab (to cycle backward).

---

## Map List View



The Sample Mapper section in the upper pane of the Sample Map Editor window, map list view. In Map List view, the sample map list (upper pane in the Sample Map Editor) displays ten columns of data for each sample. You can sort the list by the values in any column by clicking the column label: once to sort up, again to sort down.

You can edit the data values in all of these columns, except where noted:

- **L**: Left end of the key zone; i.e. the lowest MIDI note number that will play the sample. See below, **Trp**.
- **R**: Right end of the key zone; i.e. the highest MIDI note number that will play the sample.

**L** and **R** define a contiguous set of MIDI notes (keys on a MIDI keyboard) that will play the sample. This is clearly displayed in Map Keyboard mode (see below).

- **Name**: Name of the sample file (without the file extension). You cannot edit this value.
- **Trp**: Sample transpose value; i.e. the number of semitones the sample's **Root** value must be lowered or raised to reach its **L** value. **Trp = L - Root**. If you edit the **Trp** value, the **Root** value is automatically adjusted, but the **L** value stays the same.
- **LVel**: Low end of the velocity range; i.e. the lowest velocity that will play the sample.
- **HVel**: High end of the velocity range; i.e. the highest velocity that will play the sample.

**Lvel** and **Hvel** define a range of velocities that will play the sample. You can use **Lvel** and **Hvel** to enable the same MIDI note to play different samples, depending on the note's velocity (volume).

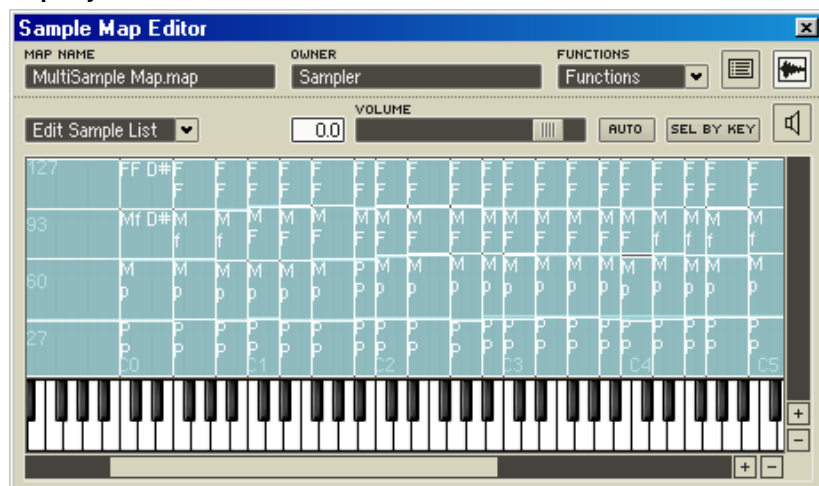
- **Root:** The MIDI note number that will play the sample at its original (intended) pitch. (See above, **Trp.**)
- **Data:** You cannot edit this value.
- **File:** You cannot edit this value.
- **Location:** The location on your hard drive where the file is located. You cannot edit this value.

---

**Note:** When the key or velocity ranges of different samples overlap, you can see which sample is given priority in Map Keyboard view. In general, the most recently added sample always gets highest priority.

---

## Map Keyboard View



The Sample Mapper section in the upper pane of the Sample Map Editor window, keyboard view

In Map Keyboard view, the sample map keyboard display (upper pane in the Sample Map Editor window) contains a two-dimensional region representing each sample in the map and a keyboard graphic along the bottom. The position, as well as the horizontal and vertical size, determine which MIDI note events cause a sample to be played. The vertical position determines the velocity range and the horizontal position determines the pitch range. You can move

the regions around with the mouse when it shows a four-direction cursor, and you can move any edge when the mouse shows a two-direction cursor.

You can select multiple regions for editing, in which case all cursor activity will apply to relatively all regions.

Regions can be made to overlap, but this only is possible for easier positioning of a region. It is not possible to playback two samples at the same time. Therefore overlaps should always be avoided.

## Functions List Box

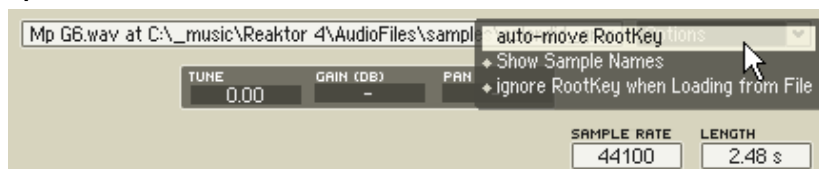
### Remap to Single Keys

**Remap to Single Keys** causes each sample in the map (regardless of which samples are selected) to be mapped to one key starting at the left-most key used in the current sample map. The order of samples in the original map is preserved.

### Set Transpose to Null for All

**Set Transpose To Null For All** resets the transpose amount of each sample to zero and changes the root key accordingly. (The root key is always the left-most key minus the transpose value.) Setting transpose to zero has the effect of making the left-most key in each region play the sample at its natural pitch. You can best see its effect by using it in the List view.

## Options List Box



Below the sample map display is a text box for displaying the name and location of the selected sample. The options drop-down menu to the right of the text box has the following entries:

- When the **Show Sample Names** option is active, the names of the samples are displayed in the graphic sample map display.
- When **Ignore Root Key when Loading** is active, the root key information saved in the sample file header is ignored.
- When **Single Key Mode** is active (meaningful in list display mode only), changing the left-split note will automatically change the right-split note to the same value. This is an editing convenience.

## Auditioning Sample Files




The Sample Map Editor enables you to audition (pre-listen to) a sample file, once it has been loaded:

1. Select a loaded sample to audition.
2. Click on the **Play** button (speaker icon) to start/stop auditioning. Use the Volume fader to set the volume.
3. If you turn **Auto** on, auditioning starts automatically when you select a sample.

## Loop Editor

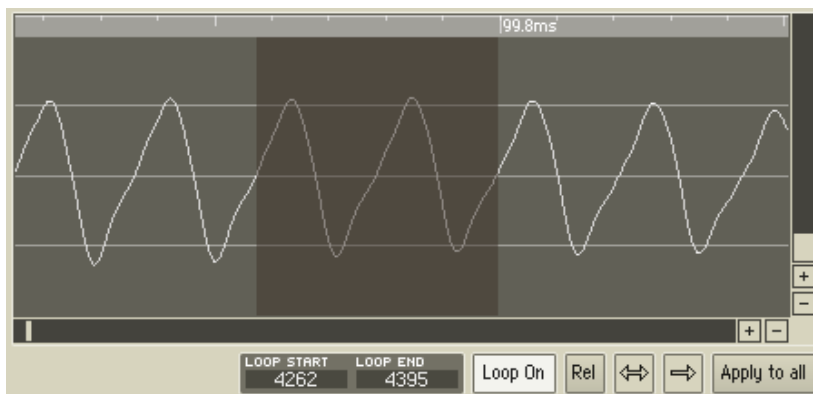
The **Loop Editor** (located in the lower half of the Sample Map Editor) enables you to set and edit loops for individual samples. (If the Loop Editor window

is not displayed, click the  **Show/Hide Loop Editor** button (waveform icon) in the upper-right corner of the Sample Map Editor.) The reddish-brown area in the sample waveform display specifies the portion of the sample that will be looped.

To change the loop start and end points, drag the start/end edges as desired. To move the entire loop, drag it (by the middle) as desired. Note that you might need to change the zoom level of the waveform display to see the entire loop. To do this, use the Zoom In/Out buttons in the lower-right corner of the display.

Note that the loop start and end points are displayed in sample numbers in the Loop Start and Loop End boxes below the waveform. But you cannot change the start/end points by editing these boxes.





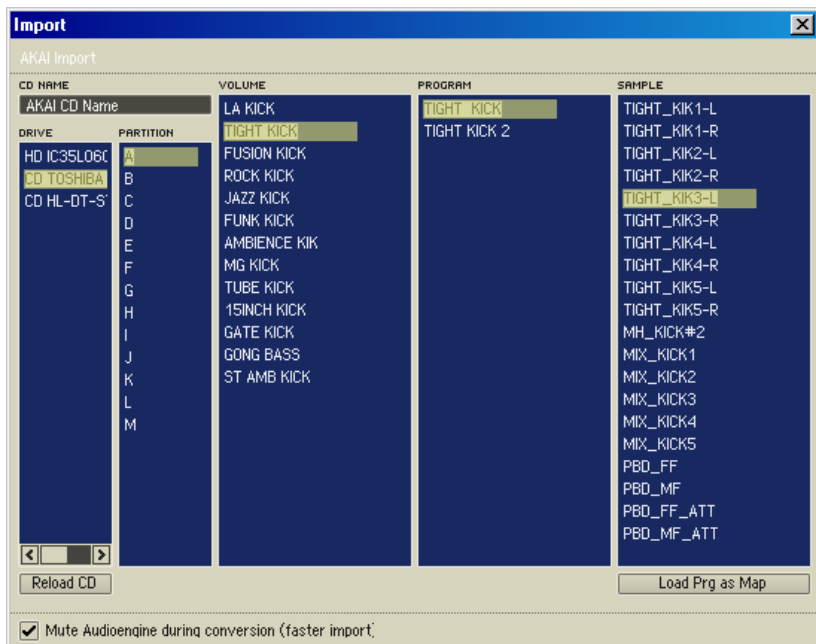
The Loop Editor section in the lower pane of the Sample Map Editor window

The buttons below the waveform display control the looping properties for the selected sample:

- **Loop On:** Enables/disables looping for the sample.
- **Rel:** Enables/disables Loop in Release mode. When this mode is enabled, samples continue looping after the gate signal has turned off (i.e. during the release stage of an ADSR envelope).
- **<->:** Enables/disables Alternating Loop mode. When this mode is enabled, the sample loop reverses direction at each end: The loop plays forward, then backward, then forward, then backward, and so on. This often produces a smoother loop with less of a hitch at its looping point.
- **->:** Enables/disables Reverse Playback mode. When this mode is enabled (i.e. when the arrow points to the left), the entire sample (not just the loop) plays backward.
- **Apply to All:** Applies all the current loop settings (i.e. of the four buttons above) to all samples in the map.

## 16.4. Akai Import

You can import Akai files from the Sample Map Editor or from a sampler module's panel (not structure) context menu. Select **Akai Import** to open the Akai Import window that shows the contents of an Akai-formatted CD loaded in your CD drive. If the CD contents don't appear, click the **Reload CD** button.



### Akai Import window

The Akai Import window displays all of the Akai CD's partitions, volumes, programs, and samples, and enables you to convert selected Akai samples and programs to REAKTOR Map files (\*.map). Converted files are stored in the folder specified by **Imported Files (Akai)** in the REAKTOR Preferences dialog (Directories page).

When loading or converting to a Map file, REAKTOR preserves the following sample information:

- Sample data
- Loop points
- Root key
- Pan

The following information is not preserved during conversion:

- Filter settings
- Envelope settings
- Velocity splits
- Gain

It is also possible to load Akai samples and programs directly into a sampler module by clicking the **Load Prg as Map** button (in the Akai Import window). You can save these samples as a Map file by selecting **Save Map** from the Edit Sample List box in the Sample Map Editor. Or you can save them with the ensemble by turning on **Store Map with Module** in the module's Properties dialog (Function page).

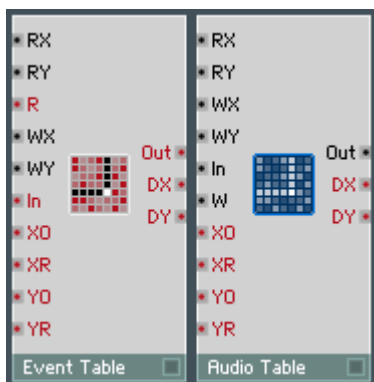
To speed up importing, turn the **Mute Audioengine During Conversion option** on. Note that doing this will stop all REAKTOR audio output until the importing is finished.

## 17. Table Modules

The Table modules allow for very flexible handling of events and audio data. The Table modules can be used to design oscillators, LFOs or waveshapers by drawing your own waveforms with the mouse. Or you can crossfade between wavetables and create envelopes with curve shapes drawn by hand or with countless breakpoints. The Event Table can be used as a sequencer for outputting gate and pitch values or for controlling any parameter in a REAKTOR Ensemble.

### 17.1. Properties

The Event Table and Audio Table modules have an extensive Properties window which is identical for both kinds of module. Some general properties which are already described for many other modules are not covered again here.



The Event Table module and the Audio Table module

## Function Page

**Properties**

**Event Table - Properties**

**LABEL**  
Event Table

**STATUS**  
☐ Mono ☐ Mute

**INTERPOLATION**  
None

**CLIP/WRAP XY**  
Wrap

☒ Backup Data With Module

**FILE**  
<empty>

**CLIENTS**  
-

**X SIZE**  
empty

**Y SIZE**  
empty

**VALUE**

MAX	STEPSIZE	DEFAULT
0	0	0
MIN	NUM STEPS	
0	0	

**DISPLAY UNITS**  
-

**X UNITS**  
-

**Y UNITS**  
-

SAMPLES/SEC	BPM	TICKS/BEAT
0	0.00	0
SAMPLES/TICK	NUM. OF BEATS	
0	0	

Properties dialog of a Event Table module, Function page

## Interpolation

- **None:** No interpolation happens when reading between cell values. Only the integer part of values arriving at the **RX** and **RY** inputs are used. The result is a stepped output signal even when the input position changes smoothly. The display looks stepped like a bar chart in the 1D-modes.
- **X:** Interpolation between values is only used on the X-axis. Fractional values at the **RX** input are used to compute smooth transitions between table cells.

- **Y:** Interpolation between values is only used on the Y-axis. Fractional values at the **RY** input are used to compute smooth transitions between table cells.
- **XY:** Interpolation between values is used in both the X and Y-axis. The full precision of fractional values at the **RX** and **RY** inputs is used to compute smooth transitions when reading between table cells.

### Clip/Wrap XY

- **Clip:** When reading beyond the end of the table you get the value of the last cell. Reading before the start of the table you get the value of the first cell.
- **Wrap:** When reading beyond the end or start of the table it continues at the other end of the table as if it were connected in a circular loop.

### Backup Data with Module

Activate this option to save the data in the table within the Ensemble, instrument or macro file.

### File

The data in the table can be read from or stored to a file with the **Load** and **Save** buttons. The **New** button creates a new, empty table. The Audio Table and Event Table modules can read the following file formats:

- table files (\*.ntf)
- audio samples (\*.wav or \*.aif)
- plain text (\*.txt) containing numbers separated by spaces (Text files are treated as one row of data, so Y-Size is always 1.)

The name of a loaded file will be displayed in the File Name field.

It is possible to use a text editor to create a Table file. Just enter values for the X axis in a row with spaces between the values. Save the file with the file extension \*.txt. It is not possible to create values for the Y-axis using a text file (Y always equals 0).

You can save the data from a table as a file for reuse in other Table modules. If the same file is loaded into more than one Table module in the same Ensemble, the data in this file will be shared between these modules. Modifying the table content in one module affects all other modules. If all modules display the content of the same table cells, any modification of the values is visible in realtime in the panel graphs of all table modules.

The **Clients** field shows the number of Table modules in the Ensemble which share the same Table file.

## X Size/Y Size

With the **Set**-Button you can define the size of the table storage. The first field is for the number of cells on the X-axis (the width of the rows), the second for the number of cells on the Y-axis (height of the columns). Changes to the numbers of cells will not become valid until the **Apply** button is pressed.

---

**Note:** If you reduce the number of cells, then any data contained in the removed cells will also be deleted.

---

## Value

**Min**, **Max**, **Stepsize** and **Num Steps** work just like they do with knobs and faders.

The **Default** value is used to initialize cells when creating or enlarging a table or when cutting a selection from it. The **Default** value is also important in the display because it appears as a distinct color (normally black). **Default** is commonly set to 0.

## Display Units

When editing the table in Draw mode, the current value is displayed in the graph's status bar in a format according to the **Display Units** setting.

- **Numeric:** Standard format for numbers in any range.
- **MIDI Note:** The value is rounded to the nearest integer and displayed as the equivalent MIDI Note number. For example, 60 is C3 and 58 is A#2.
- **% (percent):** The range 0...1 is displayed as 0...100%. For example, 0.5 becomes 50% and 2 becomes 200%.

## X Units

Sets the units which are used to measure the horizontal cell position in the table. The following units are available:

- **Index:** This is the default unit. The cells are numbered with integer values (0, 1, 2 ... n).
- **[0...1]:** The first cell has the position 0, the last the position 1. The position of the cells inbetween are computed by their relative location in the table as a fractional value between 0 and 1.

- **Milliseconds:** The position of a cell will be computed as time in milliseconds (ms), depending on the sample rate entered in the **Samples/Sec** field below. This unit is especially interesting in the Audio Table module for moving inside an audio sample which was recorded in real time.
- **Tempo Ticks:** The position of a cell will be computed in ticks of a tempo clock. This option is especially useful in the Audio Table module with a rhythmic piece of audio loaded for which you know the BPM (Beats per Minute) tempo. The **Ticks/Beat** field defines how many ticks make up a beat (usually 24).

When **X Units** is set to **Milliseconds** you can adjust the sample rate of the data:

- **Samples/Sec:** How many cells correspond to one second of time. The sample rate of a loaded Wav or Aiff file will automatically appear here.

When **X Units** is set to **Tempo Ticks** you can also adjust the following values:

- **Samples/Tick** :How many cells in a subdivision of a beat.
- **BPM:** Tempo in beats per minute.
- **Ticks/Beat:** The subdivision of a beat. For REAKTOR's Master Clock this is 24.
- **Number of Beats:** The length of the data measured in beats (usually 4 or 8 for a smoothly looped audio beat).

When changing one of these values, the others are automatically recomputed to match the length of the data and sample rate.

## Y Units

Sets the units which are used to measure the vertical cell position in the table. The following units are available:

- **Index** :This is the default unit. The rows are numbered with integer values (0, 1, 2 ... n).
- **[0...1]:** The first row has the position 0, the last the position 1. The position of the rows inbetween are computed by their relative location in the table as a fractional value between 0 and 1.



## Appearance Page

**Properties** ✕

**Audio Table - Properties**

**LABEL**  
Audio Table

**VISIBLE IN VIEW A AND B**

☒ Picture ☒ H Scroll Bar  
☒ V Scroll Bar

**SIZE X**  
384 Pixels

**SIZE Y**  
128 Pixels

**GRAPH**  
Line

**VIEW PARAMETERS**

☒ X Auto Fit ☒ Y Auto Fit ☒ Value Auto Fit

**ALIGNMENT**

☒ X ☒ Y

**GRID**

Grid	Step	Sz1	Sz2	Sz3	Sz4
X	-	-	-	-	-
Y	-	-	-	-	-
Value	-	-	-	-	-

**ENABLE GRID**

☐ X ☐ Y ☐ Value

**VIEWS**  
A B AB COPY A > B COPY B > A

**VISIBLE**

☒ Label ☒ Value ☒ Visible ☐ Small Label/Value

Properties dialog of a Event Table module, Appearance page

### Visible in View A and B

The settings in this area are global module settings and always apply to both panel views, A and B.

- **Picture:** Tick this option to see the table display in the panel.
- **H(orizontal) Scroll Bar:** Tick this option to see a scrollbar under the table graph.
- **V(ertical) Scroll Bar:** Tick this option to see a scrollbar to the right side of the table graph.

## Size X/Size Y

Enter here the size of the Table display within the Panel. The display size is entered in pixels.

## Graph Format

This drop-down list selects how the values are displayed in the Table module's panel graph. You can choose between four modes:

- **Pixel:** Values are drawn with a horizontal line. If you set no interpolation for the X axis the lines look like small faders.
- **Line:** Values are drawn with a horizontal line and vertical lines are also drawn to connect the values.
- **Bar:** Values are drawn with a horizontal line, vertical lines are drawn to connect and the area underneath is filled with color.
- **2D Color:** This mode is for viewing more than one row at a time. This is what you need to see all of a 2-dimensional table (Y-Size > 1). The value of each cell is displayed as the color of the corresponding rectangle. The rows are numbered from top to bottom, the columns always from left to right.
- **2D Curve:** Similar to the setting **2D Color**, but in contrast to it you can edit multiple rows at the same time by drawing their curve shapes. While the **2D Color** setting offers the bird's eye view on the Table data, **2D Curve** allows the front view.
- **Solid:** Like the setting **Bar**, but without the outline.

## View Parameters

- **X Auto Fit:** When this option is ticked all cells in the X direction are always displayed in the graph. The number of displayed cells is the same as that indicated by **X Size** in the **Table** tab.
- **X Alignment:** When **Auto Fit** is off this slider controls how smaller regions of the data are selected. When the slider is in the left position the cell selected with the **X0** input appears at the left edge of the display. With the slider centered, the selected cell appears in the middle. When the slider is in the right position the last cell visible at the right edge of the display is the one before **X0**.
- **Y Auto Fit:** When this option is ticked and **2D** mode is selected, all cells in the Y direction are always displayed in the graph. The number of displayed cells is the same as that indicated by **Y Size** in the **Table** tab. **In display modes other than 2D, Auto Fit has no effect.**

- **Y Alignment:** When **Auto Fit** is off **Value Auto Fit:** When this option is ticked the display's value range in Pixel, Line and Bar mode is the same as the range set with **Min** and **Max** on the Table tab. In 2D mode **Auto Fit** has no effect.

## Grid

The settings are the same for X, Y and Value Grid:

- **Enable Grid:** When these boxes are ticked vertical lines (**X Grid**) or horizontal lines (**Y Grid** in 2D display mode or **Value Grid** in the Pixel, Line or Bar display mode) will be displayed in the panel graph.
- **Grid Step:** The value entered here relates to units set for X and Y on the Table tab. It sets the spacing of the grid. If you want to set the basic resolution of the grid to one unit enter 1 here. Enter 2 to have every second unit as a grid step, or 0.5 for two grid steps per unit.
- **Size 1 ... Size 4:** There are four different sizes available for the grid lines: 1 is the finest and 4 is the thickest line size. The number you enter in these boxes defines how often a line with a certain thickness is drawn (number of Grid Steps per line). Enter 1 for a line at every Grid Step, or 2 for a line at every other Grid Step, for example. Enter 0 if you do not want to use a size at all.

## Visible

The panel display options in this area can be set for panel A or B only, or for both panels, depending on what panel button is activated.

- **Label:** Tick this checkbox for displaying the module name in the panel.
- **Visible:** The graphic area is displayed on the panel when this option is enabled. It is used for viewing and editing data.
- **Value:** The status bar at the top of the Table graph is visible when this option is selected. It shows information relating to the mouse position in the graph, such as the X and Y position and the value in the table at that position. The units of the displayed values depend on the selected units in the Properties box under the **Table** tab.
- **Small Label/Value:** Tick this checkbox to set a smaller display size for the module label and value status bar in the panel.

## 17.2. Context Menu

When you (Windows XP) click the right mouse button / (OS X) hold Ctrl. and click the mouse button on a table graph, a special context menu for editing in the graph becomes visible. For ease and speed of navigation, we recommend becoming familiar with the keyboard shortcuts for editing operations. Keep in mind that the shortcuts only apply to the data in the graphs when the **Panel Lock** mode for the Instrument is active.

### Draw/Select/Control Mode

- **Table Draw Mode:** This mode allows you to enter values with the mouse. You can draw curves or modify single values by moving the mouse up and down. In 2D-mode, set the value with which to draw by selecting the menu option **Set 2D Draw Value...** or pick an existing value by Ctrl-mouse clicking on a cell in the graph.
- **Table Select Mode:** In **Select** mode, instead of drawing with the mouse to change the values, you use the mouse to select a region for modifying later. The selected data can then be worked on with the editing functions.
- **Table Control Mode:** In **this** mode Table data cannot be changed in any way. You can neither enter new data nor can you modify existing data via the panel.

### File

- **Load Data into Table, Save Table Data:** These menu options are shortcuts for the **Load** and **Save** buttons in the **File** section of the **Table** tab in the Properties window.
- **Save Table Data as...:** Save a Table file under a new name.
- **Reload Table Data:** If you have modified a table file outside REAKTOR, you can load the new version with this menu command.

### Show

- **Show All:** Zooms out fully so that the whole table is visible in the graph.
- **Show Selection:** Zooms the display so that the current selection completely fills the display.
- **Next Y (Page Down):** In Pixel, Line and Bar mode the next higher row of table cells is displayed. In 2D mode the graph is scrolled vertically one row up.

- **Previous Y (Page Up):** In Pixel, Line and Bar mode the next lower row of table cells is displayed. In 2D mode the graph is scrolled vertically one row down.

## Graph

- **Pixel Mode**
- **Line Mode**
- **Bar Mode**
- **2D Color Mode**
- **2D Curve Mode**
- **Solid Mode**

It is possible to change the display mode of the table graph directly via the context menu without overwriting the Properties settings. Read more about the six graph display modes above in the description of the **Appearance** tab in the Properties window.

## View

- **Show Read Position:** When this option is selected, a vertical line at the current read position is displayed.
- **Show Write Position:** When this option is selected, a vertical line at the current write position is displayed.
- **Show Horizontal Position Line:** When this option is selected, a position ruler above the Table graph is displayed.
- **Show Horizontal Scroll Bar:** Select this option to see a scrollbar under the table graph.
- **Show Vertical Scroll Bar:** Select this option to see a scrollbar to the right side of the table graph.

## Select

- **Select All (Ctrl+A):** Selects all currently visible data.
- **Select X All:** Selects all currently visible data in the selected rows.
- **Select Y All:** Selects all currently visible data in the selected columns.
- **Snap Selection to Grid:** With this option enabled, the edges of any selection are always adjusted to lie on the Grid, which may be much coarser than the cell size, depending on the settings in the Properties'

**Grid** tab. If the Grid's smallest line size is not visible because the display is zoomed out to display a lot of data, then the selection snaps to the smallest Grid size that is still visible.

## Process

- **Mirror X:** Swaps the data between left and right using a vertically symmetric axis in the middle of the selection.
- **Mirror Y:** In 2D mode, swaps the data between top and bottom using a horizontally symmetric axis in the middle of the selection.
- **Rotate/Add/Scale....:** Allows numerical entry for applying several mathematical operations on selected data. **Add Value....:** Allows numerical entry for adding or subtracting from the values of selected data. **Rotation:** Rotates the selection by the given amount. Cells which are moved out of the selected area on one side reappear on the other side in a circular motion. **Scale Value....:** Allows numerical entry for scaling the values of selected data. (1=100%, 0.5=50%, 2=200% ...).
- **Trim Selection:** When you apply this function to a selection of Table cells, all table cells which are not located within the selection will be cropped. The number of X and Y cells in the Table will be updated after trimming.
- **Delete Rows:** All rows which are located within a selection will be deleted. The number of Y cells in the Table will be updated after applying this command.
- **Insert Rows:** The number of rows you have defined with a selection will be added to the Table. The number of Y cells in the Table will be updated after applying this command.
- **Quantize Value to Step Size:** When this option is selected, values drawn with the mouse snap to the step size set in the table module Properties window. This is the normally used mode. Unselect this option to draw at a finer resolution (regardless of the set step size).
- **Set 2D Draw Value....:** You can enter a value which is used when you draw values in 2D mode. You can also pick a value from the View by pressing Windows XP: Ctrl+Right-click / OS X: X+click in draw mode.

## Cut, Copy, Paste

- **Copy** (Windows XP: **Ctrl+C** / OSX: **X+C**): Copies the selected data into the copy buffer.
- **Cut** (Windows XP: **Ctrl+X** / OSX: **X+X**): Cuts the selected data and puts it in the copy buffer.
- **Paste** (Windows XP: **Ctrl+V** / OSX: **X+V**): Pastes data from the copy buffer into the current selection.

## 17.3. Advanced Operation

### Draw / Select Mode

The current editing mode is displayed in a square at the top left in the status bar as a **D** or **S** to indicate Draw or Select Mode. You can toggle the mode by clicking on the square.

The Tab key also toggles the editing mode.

### Rotate

By holding the Shift key and dragging with the mouse, you can move all cells in the selected area left or right, or in all directions in 2D-mode. When everything is selected, you can drag around the whole graph.

### Add

By holding the Windows XP: Ctrl key / OS X: X key and dragging the mouse up or down, you modify the value of everything in the selected area. An amount is added to or subtracted from all the values, depending on which way you drag the mouse.

### Panel Lock Mode

There is a connection between the **Panel Lock** mode of the Instrument and the Table modules. When the **Panel Lock** mode is activated, panel operations with keyboard shortcuts like copy (Windows XP: Ctrl+C / OS X: X+C) and paste (Windows XP: CTRL+V / OS X: X+V) apply to the selected data inside the table graph instead of the module itself. So when the panel is locked, you will move the data in the module instead of the module's representation on the panel.

## 18. “Classic Modular” Macro Collection

The Classic Modular macro collection is an assortment of high-level REAKTOR building blocks that simulate classic analog modular systems. In addition to impressive sound quality and flexibility that rivals the originals, the Classic Modular macros have a number of advantages over their analog counterparts. For instance, the Classic Modular macros include sampling, granular sampling, and sophisticated sequencers, in addition to a generous selection of oscillators, filters, and modulators.

Several instruments in the REAKTOR Library were created out of the Classic Modular macros: The Green Matrix synthesizer, the Blue Matrix sequenced synth, and the Analogic Filter Box were built in part out of elementary Classic Modular macros. Take some time to listen to these instruments in order to get a feel for what the Classic Modular collection can do.

These macros have two main properties:

1. Standardized signal range from -1 to 1 for all inputs and outputs. That means that all output signals can be connected to any input without down or up scaling. This is the main difference from REAKTOR's other modules and macros in which the value range can vary depending on what type of unit a parameter controls (e.g., semitones, decibels, or milliseconds). For instance the pitch (P) input of an oscillator module expects values in the range of 0 to 127, while the amplitude (A) input expects values between 0 and 1.

Even the pitch (P) signals in this collection adhere to the standard range of 0 to 1. To convert these signals to the range of 0 to 127 please use the “0-1 to 0-127 Range Converter” macro in the “Event Processing” folder.

The only exception to the rule is the position (Pos) signal of the sequencer macros. Since its integer value addresses steps in a sequence, it doesn't make sense to limit its range to 0-1.

2. For each important parameter of a macro there is a “manual” control (usually a knob) for controlling this parameter directly along with one or two modulation amount controls. For each modulation amount control there is a corresponding input with a similar name for connecting modulation sources such as LFOs and envelopes. The incoming modulation signals are scaled by the modulation amount controls and are added to the value of the manual control. If more modulation inputs are needed, it's possible to connect a modulation mixer to the



modulation input. The corresponding modulation amount control should be turned to maximum since the actual modulation amount is set with the mixer.

There are event and audio type modulation inputs. Most modulation sources (such as LFOs and envelopes) offer both types. If other audio signals should be connected to event inputs they have to be converted to event signals first with an “A to E” module (which can be found in the “Auxiliary” folder of REAKTOR’s modules).

Some file names contain “Event” or “Audio”. The purpose of these macros is to scale, mix, invert and switch between modulation signals (such as LFOs and envelopes), and they are available in event and audio versions. Some names contain “Stereo” or “Mono”. The purpose of these macros is to mix, amplify, filter or distort ‘normal’ audio signals (in contrast to modulation signals), and they are available in stereo and mono versions.

The “Classic Modular” macros can be used to build polyphonic structures. This is one big advantage over analog modular systems, which were normally monophonic. Since the output of an instrument is monophonic, polyphonic signals within the instrument have to be converted to monophonic signals with a voice combiner module (which can be found in the “Auxiliary” folder of REAKTOR’s modules). Please note that all display modules such as meters, number displays, oscilloscopes, LEDs, etc. in the “Classic Modular” macros are connected only to the last played voice!

## **18.1. Display**

### **Number Display**

Shows the current value of the input signal. If it changes too fast, we recommend turning on the built-in peak detector with the “Peak” switch. Then the overall amplitude envelope will be displayed.

### **Simple Scope**

Displays the curve of the incoming signal just like an analog oscilloscope.

### **XY Scope**

Oscilloscope in XY mode. The incoming “X” signal determines the horizontal position of the displayed dot. The incoming “Y” signal the vertical position.

## 18.2. MIDI

### Controller

Receives the following MIDI controller messages and passes them on to its outputs: pitchbend, modulation wheel, aftertouch, and volume.

### Notes - Monophonic

Receives MIDI note messages and passes their pitch, gate and note on velocity values to the corresponding outputs in monophonic manner (i.e., only one note is present at a time). This note is sent to all voices simultaneously. The key range can be set on the panel and is independent of the key range set in the Properties dialog of the instrument, which is the key range of the “Notes - Polyphonic” macro. It is possible to have several “Notes - Monophonic” macros with different key ranges.

### Notes - Polyphonic

Receives MIDI note messages and passes their pitch, gate and note on velocity values to the corresponding outputs in polyphonic manner (i.e., the notes are assigned to different voices). The key range can be set in the Properties dialog of the instrument. All “Notes - Polyphonic” macros share this key range.

### Selective Gates - MIDI Keyboard

The gate signal of 12 continuous keys starting with the one defined by the “Lower” knob are received and passed to the 12 corresponding outputs of the macro in a monophonic manner. The gate signals are sent to all voices of the instrument simultaneously. The selected key range is independent of the key range set in the Properties dialog of the instrument.

Useful applications are to trigger different envelopes with each key or to start/stop sequencers with pressing/releasing a certain key, while the other keys outside this key range are used to play notes.

### Selective Gates - QWERTY - Lower Keys

If an instrument is selected the keys from the computer keyboard are sending MIDI notes to the instrument like an external MIDI keyboard is doing. This feature is called “QWERTY”, just like the first 6 keys on the English keyboard. This macro receives the notes of the lower key range starting with the “Z” key and ends with the “M” key and passing only their gate signals to the corresponding output of the macro in a monophonic manner. Which means that

the gate signals are sent to all voices of the instrument simultaneously. There is an output for each key. The key range of the QWERTY keys is independent of the key range set in the Properties dialog of the instrument.

Useful applications are to trigger different envelopes with each key or to start/stop sequencers with pressing/releasing a certain key, while the other keys outside this key range are used to play notes.

### **Selective Gates - QWERTY - Upper Keys**

The same as the “Selective Gates - QWERTY - Lower Keys” macro but the key range starts with the “Q” key and ends with the “P” key.

## **18.3. Mixer/Amp**

### **About Amplifiers:**

Signal generators like oscillators and samplers already have built-in amplifiers (A input). If other signals have to be amplified or attenuated the amplifier macros should be used.

To amplify normal, non-modulation audio signals there are two possible ways depending on the used modulation signal. Because of the exponential way the ear perceives amplitude changes, either the modulation signal changing the amplitude or the amplifier itself should have exponential characteristics. For example the ADSR envelope has exponential curves for the decay and release phase so it can be connected to linear amplifiers like the built in ones in oscillators and samplers. The modulation signals from the modulation sequencer have a linear characteristic and should be connected to an exponential amplifier.

To amplify modulation signals, it's best to use linear amplifiers.

### **Amp - Exponential**

Amplifier with exponential characteristic. The amplitude is set in decibels (dB).

### **Amp - Linear**

Amplifier with linear characteristic. The amplitude can be set in the range from 0 to 1.

### **Crossfade**

Crossfading module. The output signal is mixed from the both input signals.

## **Inverter**

Inverts the polarity of the incoming signal. Two modes are available: In the unipolar (Uni) mode the signal is mirrored round 0.5. This is the right mode to invert non-velocity sensitive gate signals. In bipolar (Bi) mode the signal is multiplied by minus 1. This is the right choice for inverting LFOs and other bipolar modulation sources.

## **Master Volume**

Master level control with built-in limiter to prevent the signal from clipping. Should be the last macro in the instrument. The input signals must be monophonic. Please insert voice combiners (can be found in modules/Auxiliary) to convert polyphonic signals into monophonic ones.

## **Mixer - Simple**

Simple 4-channel mixer for audio signals, which aren't used for modulation. To mix modulation signals, please use the Modulation Mixer.

## **Mixer - Studio**

8-channel mixer with one post fader effect send, one pre fader effect send, panning, and an on/off button for each channel. The input signals must be monophonic. Please insert voice combiners (can be found in modules/Auxiliary) to convert polyphonic signals into monophonic ones.

## **Modulation Mixer**

3-channel mixer for modulation signals with switches for inverting the incoming signal. If a higher resolution at small values of the knobs is needed, the characteristic of the knobs can be switched from linear to exponential with the “exp” button.

Since the actual modulation amount is set with this mixer, the modulation amount control of the macro that the mixer is connected to should be set to maximum.

## **Modulation Matrix - Mixer**

8x8 mixer matrix for modulation signals. The columns correspond to the inputs, the rows correspond to the outputs of the matrix. Each row defines a mix of the modulation signals present at the columns. The mix is passed to the corresponding output. If a higher resolution at small values of the knobs is needed, the characteristic of the knobs can be switched from linear to exponential with the “exp” buttons.

Since the actual modulation amount is set with this matrix mixer the modulation amount control of the macro the mixer is connected to should be set to maximum.

### **Modulation Matrix - Switch**

An 8x8 switch matrix for audio modulation signals. The rows correspond to the inputs, the columns correspond to the outputs of the matrix. One modulation signal out of eight can be selected. This signal is passed to the corresponding output without scaling.

### **Panner**

The input signal can be positioned between the two outputs.

### **Scanner**

The 8 inputs are scanned in dependence of the scan position. If a scan position is set between two inputs the output signal is obtained by crossfading between these two.

## **18.4. Oscillator**

### **Geiger - Counter**

Generates impulses or pulses at random intervals, much like a Geiger counter radiation particle detector. The average rate and randomness of the impulses/pulses can be set.

### **Noise**

Noise generator offering four different noise types:

**White noise:** All frequencies have the same amplitude.

**Pink noise:** High frequencies are damped with 3 dB per octave. A filter sweep with a bandpass filter results in a signal with a constant amplitude.

**Coloured (filtered) noise:** The colour can be set with the “Colour”-knob.

**808:** Noise source used in the legendary TR-808 drum machine to synthesize the hihats and cymbals. Consists of 6 detuned pulse oscillators.

## **Oscillator - Symmetry**

Oscillator with symmetry/pulse width modulation for all available waveforms: bipolar ramp pulse (similar to sawtooth), bipolar pulse, normal pulse, triangle/sawtooth and parabolic (similar to sine wave). Please note that the two symmetry/pulse width modulation inputs have different characteristics: The first has a linear characteristic, which sounds good for LFOs. The second has an exponential characteristic, which sounds good for envelopes.

## **Oscillator - Sync**

Versatile oscillator with hard and soft synchronization, phase modulation and frequency modulation. Available waveforms are sawtooth, pulse, triangle, sine and impulse. On hard synchronization the oscillator restarts at the set phase. On soft synchronization the oscillator plays back its waveform in reverse direction. This results in a more gentle “sync” sound.

## **Random**

Random level sample + hold generator. Random numbers are generated in the set rate and held until the next number is generated. If the “Rmp” switch is turned on connecting ramps are generated between successive numbers.

## **18.5. Sampler**

Samplers play back samples, which are included in the sample map. A sample map editor can be found on the “gearwheel” page of the samplers’ properties. The properties can be opened with a double-click on the sample display. The “Select” knob of the samplers selects a sample of the sample map.

For playing back beat loop samples the “Classic Sampler” or the “Beat Loop” sampler should be used. The advantage of the “Beat loop” sampler is that the tempo and the pitch of the playback can be set independently. If this isn’t wanted the “Classic Sampler” should be used. Like all main parameters the loop length of these samplers can be modulated. To assure that the loop length is always multiples of a suitable musical length like 1/4 notes, one bar etc the loop length can be quantized to the step size set with the “LL Q” knob. A step size of zero turns off the quantization. Similar quantization features are available for the sample start position, loop start position and position offset (only for the “Beat Loop” Sampler). Please note that for the “Classic Sampler” the loop playback has to be turned on in the sampler’s properties while the “Beat Loop” sampler is always in loop mode.

## Beat Loop

Sampler specialized in playing back beat loop samples.

Synchronises any beat-loop sample, regardless of the original tempo, to a clock source that is connected to the “Clk” input of the macro. The playback pitch of the sample is independent of the tempo. The sample is played back in a permanent loop.

Beside the sample selection and playback pitch following parameters can be set and modulated: start position, loop start position, loop length and position offset. The unit of the corresponding controls is set with the “Unit” knob in 16th notes.

The macro has two position outputs to drive sequencers. The position events at the “Pos” output are generated at the beginning of the 16th notes and should be used to drive sequencers, which do not modulate the parameters of the “Beat Loop” sampler itself. The position events at the “Pos\*” output are generated at the beginning of each grain, which is a 32nd note before the next 16th note. Since all of the parameters of the “Beat Loop” macro except the start position are sampled and held in that moment, this output should be connected to sequencers which modulate one of these parameters.

## Classic Sampler

The “Classic Sampler” plays back samples the “old fashioned” way, linking playback pitch and speed. The Classic Sampler features frequency modulation and the possibility to reverse the playback direction with modulation signals. In the sample map editor, the loop feature can be activated for each sample individually.

The start position, loop start position and loop length are set in 1/128 of the sample length. That means that if the sample is one bar long then multiples of 8 would address 16th notes positions.

## Resynth

Sample resynthesizer with independent control over pitch and playback speed, which is specialized in playing back samples without beats. The resynth sampler cuts the sample in small pieces called “grains”. During playback these grains are played one after the other. The “granularity (Granu)” knob determines the numbers of grains in a certain time. To reduce glitches the grains can overlap each other. The fade time between two overlapping grains can be set with the “Smooth” knob.

The start position, loop start position and loop length are set in 1/128 of the sample length. That means that if the sample is one bar long then multiples of 8 would address 16th notes positions.

## 18.6. Sequencer

This folder contains a selection of macros to build powerful sequencers with. There are three classes of macros: clock generators, clock modifiers and the actual sequencers, which are driven, by the clock. The clock generator creates events in a certain rate. With each event the value is incremented by 1. This value corresponds to a certain position within a sequence stored in a sequencer. That is why these events are called “position events”.

The clock modifiers manipulate the stream of position events and can be inserted between the clock and the sequencers.

### Global Clock

Sends out the position events generated by the global clock. This clock can be started/stopped by the corresponding buttons in the toolbar of REAKTOR. The time resolution of the events can be set on the panel. Another output signal is the clock gate. It is zero when the clock is stopped and one when the clock is running, i.e. after Start or Continue. Some sequencers need this signal to get initialised, the note sequencer needs it to prevent note hanger on sequencer stop.

### Position Delay

Position event delay, which can be modulated to achieve a shuffled time resolution.

### Position Looper

Loops the incoming position events. Unlike the built in loop function of the sequencer macros, the loop start and length can be modulated by a modulation signal. Also a different mode is implemented called “Freerun”. In this mode the looper only jumps back to the loop start if it reaches the loop end. In the “Hardsync” mode the looper folds every incoming position event in the loop range so the looper would jump into the new loop immediately once the loop start is shifted by X steps. In “Freerun” mode it would take X steps to get into the new loop.

### Position Offset

Adds an offset to the incoming position events to shift the read out position within a sequence.



## **Sequencer - 1x Notes, 4x Mod, 8x Trigger**

A combination of a note sequencer, a 4 track modulation sequencer and a 8 track trigger sequencer sharing the same sequence number and global controls like the loop bar, edit bar, view bar and the global functions like copy, paste, clear and record start/stop. More information on the sequencers can be found in the explanations of the “Sequencer - Note”, “Sequencer - Modulation 4x” and “Sequencer - Trigger 8x” macro. Note that the Blue Matrix synth in the REAKTOR Library is driven by this sequencer. Please see the Blue Matrix documentation for a detailed overview of the sequencer in action.

## **Sequencer - Classic Step**

A classic step sequencer with 16 steps. Unlike the other sequencers in this collection, the classic step sequencer is built with faders, with one fader for each step. This sequencer is the one to choose if the values of the steps should be controlled remotely via a midi fader box or other midi controllers.

## **Sequencer - Modulation 4x**

Sequencer for playing back 4 parallel modulation signals. These signals can be used to modulate any synthesis parameters, such as oscillators and filters, just as LFOs and envelopes do.

A click on the “View” button toggles between the “all” and “solo” view. In “all” view all channels are visible at once. In “solo” view only one channel is visible, displayed with a higher vertical resolution. The “Select” bar (the second vertical bar from the left) selects the channel in “solo” view.

The “Seq” knob selects one of 128 different sequences so each snapshot can have its unique sequence number. The length of a sequence is 768 steps which equals 8 bars in a 96th notes resolution or 48 bars in a 16th note resolution. The numbers of sequences and the length can be set in the Properties dialog of the sequence display. The variables can be found on the “gearwheel” page of the properties. The “X” variable is the length and “Y” is the number of sequences. Since the last 8 ( $4 \times 2$ ) sequences are used as copy/paste and undo buffers the number of sequences must be at least 12 ( $4 \times 3$ ).

Please note that the sequencer doesn’t have an edit buffer. All changes are stored immediately. If you want to create different variations of a sequence, you have to make a copy of the sequence using the copy and paste buttons before editing, or else the original sequence will be altered.

On the “Eye” page of the properties the time grid of the sequencer can be adjusted to the time base of the incoming position events (“Pos”-In) from the connected clock. If they have a 96th notes resolution the “Grid step” should

be set to 6, if they have a 16th notes resolution it should be set to 1. The time grid will be visible in the sequencer, depending on the grid step value.

There are three horizontal bars: the edit bar, the loop bar and the view bar. To alter the size of these bars, their left or right ends can be clicked and dragged. Clicking and dragging in the middle functions like a scroll bar. A click beside the bars will cause them to scroll one length to the left or to the right. The edit bar determines the sequence range the following functions can be applied on: copy/paste/cut/insert, randomization (Rand), quantization (Quant), ramp, clear, and recording (Rec). The loop bar determines the sequence range, which is looped during playback. The view bar determines the visible range of the sequence. A quantization can be set for both the edit and loop bars with the "Bar /" control.

Information on the functions can be found in the mouse-over hints of the function buttons. Please note that they are applied to all visible channels.

The modulation signals connected to the "Mod" inputs of the macro can be recorded. The "recE" buttons of the channels has to be turned on to enable the recording. The recording starts when the "Rec 1/0" is pressed. Please note that the recording is bound to the range defined by the horizontal edit bar. If the "1 shot" button is on, the actual recording will start once the first modulation event is received. Recording will stop after the locator has passed once through the edit region.

## **Sequencer - Note**

The Note Sequencer is used to sequence notes with a standard piano-roll style editor. The sequencer consists of two data fields. The upper is for the actual notes, displaying the notes in piano-roll style: The horizontal direction represents the time, the vertical the pitch of the notes. A note starts when the pitch graph exceeds an adjustable threshold and ends when it falls below the threshold. The threshold is set with the second vertical bar of the upper data field. The lower data field is for generating re-trigger events in the notes set in the upper data field and for defining the velocity of the notes.

The "Seq" knob selects one of 128 different sequences so each snapshot can have its unique sequence number. The length of a sequence is 768 steps which equals 8 bars in a 96th notes resolution or 48 bars in a 16th note resolution. The numbers of sequences and the length can be set in the Properties dialog of the data fields and must be the same for both. The variables can be found on the "gearwheel" page of the properties. The "X" variable is the length and "Y" is the number of sequences. Since the last two sequences are used as copy/paste and undo buffers the number of sequences must be at least 3.

Please note that the sequencer doesn't have an edit buffer. All changes are stored immediately. If you want to create different variations of a sequence, you have to make a copy of the sequence using the copy and paste buttons before editing, or else the original sequence will be altered.

On the "Eye" page of the properties the time grid of the sequencer can be adjusted to the time base of the incoming position events ("Pos"-In) from the connected clock. If they have a 96th notes resolution the "Grid step" should be set to 6, if they have a 16th notes resolution it should be set to 1. The time grid will be visible in the sequencer, depending on the grid step value.

There are three horizontal bars: the edit bar, the loop bar and the view bar. To alter the size of these bars, their left or right ends can be clicked and dragged. Clicking and dragging in the middle functions like a scroll bar. A click beside the bars will cause them to scroll one length to the left or to the right. The edit bar determines the sequence range the following functions can be applied on: copy/paste/cut/insert, randomization (Rand), quantization (Quant), clear and recording (Rec). The loop bar determines the sequence range, which is scanned and looped during playback. The view bar determines the visible range of the sequence. A quantization can be set for both the edit and loop bars with the "Bar /" control.

Information on the functions can be found in the mouse-over hints of the function buttons.

The pitch and gate signals connected to the "P" and "G" inputs of the macro can be recorded. To do so the "recE" button has to be turned on to enable the recording. The recording starts when "Rec 1/0" is pressed. Please note that the recording is bound to the range defined by the horizontal edit bar. If the "1 shot" button is on the actual recording will start with the advent of the first note to be recorded and will stop after the locator has gone thru the edit bar region once.

## **Sequencer - Simple Modulation**

A simple modulation sequencer. Similar to the "Sequencer - Modulation 4x" sequencer but with only one channel and less features.

## **Sequencer - Trigger 8x**

Sequencer for playing back 8 parallel trigger channels. These triggers can be used to trigger envelopes, samplers, drum synthesizers etc.

A click on the "View" button toggles between the "all" and "solo" view. In "all" view all channels are visible at once. In "solo" view only one channel is visible, displayed with a higher vertical resolution. The "Select" bar (the second vertical bar from the left) selects the channel in "solo" view.

The “Seq” knob selects one of 128 different sequences so each snapshot can have its unique sequence number. The length of a sequence is 768 steps which equals 8 bars in a 96th notes resolution or 48 bars in a 16th note resolution. The numbers of sequences and the length can be set in the Properties dialog of the sequence display. The variables can be found on the “gearwheel” page of the properties. The “X” variable is the length and “Y” is the number of sequences. Since the last 16 ( $8 \times 2$ ) sequences are used as copy/paste and undo buffers the number of sequences must be at least 24 ( $8 \times 3$ ).

Please note that the sequencer doesn't have an edit buffer. All changes are stored immediately. So if you want to do versions of a sequence you have to make a copy of the sequence using the copy and paste buttons before editing otherwise the “original” sequence is altered.

On the “Eye” page of the properties the time grid of the sequencer can be adjusted to the time base of the incoming position events (“Pos”-In) from the connected clock. If they have a 96th notes resolution the “Grid step” should be set to 6, if they have a 16th notes resolution it should be set to 1. Then an appropriate time grid is generated.

There are three horizontal bars: the edit bar, the loop bar and the view bar. To alter the size of these bars their left or right ends have to be clicked and dragged. To move them the middle of the bars have to be clicked and dragged. A click beside the bars lets them jump one length to the left or to the right. The edit bar determines the sequence range the following functions can be applied on: copy/paste/cut/insert, randomization (Rand), quantization (Quant), clear and recording (Rec). The loop bar determines the sequence range, which is scanned and looped during playback. The view bar determines the visible range of the sequence. For the edit bar and the loop bar a quantization can be set with the “Bar /” control.

Information on the functions can be found in the mouse over hints of the function buttons. Please note that they are applied to all visible channels at once.

The trigger signals connected to the “Trig” inputs of the macro can be recorded. To do so the “recE” buttons of the channels has to be turned on to enable the recording. The recording starts when the “Rec 1/0” is pressed. Please note that the recording is bound to the range defined by the horizontal edit bar. If the “1 shot” button is on the actual recording will start once the first trigger event is received. Recording will stop after the locator has passed once through the edit region.

## 18.7. LFO, Envelope

### Envelope - ADSR

Envelope generator with the classic attack-decay-sustain-release characteristic.

### Envelope - Decay

Envelope generator with the decay characteristic.

### Envelope - One-Ramp

Envelope generator which generates a ramp between the set start and end point within an adjustable time. The shape of the ramp can be switched from exponential to linear with the “lin” button. The values of the start and end point are sampled and held in the moment the envelope is triggered if the “s/h” buttons are turned on.

### Envelope Follower

The envelope follower generates an envelope in dependence on the amplitude of the input signal.

In “Peak” mode the output signal follows the amplitude peaks of the input signal. The input signal is rectified and smoothed by an adjustable release time. The attack time is zero.

In “Roots means square (Rms)” mode, the output signal follows the loudness of the input signal. That means for instance that short impulsive signals will not enter the output signal so much as in “Peak” mode since the shorter a signal gets, the smaller is its loudness. Technically the RMS value is the square root of the average of the squared values over the specified time interval.

### LFO

Low frequency oscillator providing the following waveforms: slow random, sine, triangle, and pulse. The “Wave” control scans through them; intermediate positions between two successive waveforms can be set. The symmetry or pulse width of the waveforms is set with the “Width” control. A rising sawtooth is a triangle wave with the “Width” set to 1. For a falling sawtooth the “Width” control must be set to minus 1.

There are three different modes for setting the speed of the LFO, which is set with the “Unit” switch in the upper left corner of the macro. In “P” mode the “Speed” control changes the pitch of the LFO in semitones. In “bpm”

mode the speed is set relative to the beats per minute (bpm) of the global clock of REAKTOR. In “pos” mode the speed is set relative to the frequency of the incoming events at the “Pos” input of the macro. The last mode should be selected if the LFO should sync to a clock, which is independent from the global clock. Then the position events of this independent clock must be connected to the “pos” input of the macro. The “Div” control divides the speed of the global clock and the measured speed of the incoming events at the “pos” input of the macro. E.g. in “bpm” mode the “Div” control sets the unit of the “Speed” control in 1/96 notes. 6 corresponds to 16th notes, 12 to 8th notes, 24 to quarter notes etc. In the “Pos” mode this correspondence applies if the incoming position events have a 1/96 notes resolution.

The LFO can be synced to an external signal or if the “Unit” switch is set to “bpm” or “pos” to the song position. When the synchronization occurs the LFO restarts at the phase set with the “Phase” control.

### **Sample and Hold**

With incoming events at the “TE” input or at clock edges present at the “C” input, the current value of the incoming signal is sampled and held until the next sample is taken. It’s possible to adjust the conditions for sampling from the panel.

### **Triggered Random**

Random number generator that transmits a random number each time a clock edge in the input signal is detected. The “Edge” switch defines on which occasion a random number is triggered. If set to “+” the trigger occurs when the input signal rises above zero. If set to “-” the trigger occurs when the input signal falls under zero. If set to “both” the trigger occurs in both cases.

## **18.8. Filter**

### **3 Band Filter**

Versatile 3 band equalizer. Each band can be muted to achieve different filter curves including highpass, bandpass, lowpass and notch.

This filter can be used to simulate the “kill” filters of disc jockey mixers.

### **Bandsplit**

Splits the incoming signal in high, mid, and low frequency bands for further processing. Mixing these bands would result in the unfiltered signal without any coloration.

## **Comb**

Comb filter produces flanger and chorus effects. A comb filter mixes the input signal with the delayed input signal, resulting in a frequency spectrum that resembles a comb with multiple sharp peaks and valleys. At multiples of the set filter frequency ( $1/\text{delaytime}$ ), there are resonant peaks while at multiples of .5, 1.5, 2.5 etc of the filter frequency, the sound is cancelled out.

The amount of feedback (filter resonance) can be set with the corresponding knob. The “GainC” knob determines how much an increase of the feedback decreases the output level of the filter, which is especially helpful if the feedback is modulated. The “Ex” switch activates the external feedback path. The delayed signal is sent out at the “Send” output of the macro so it can be processed by filters and other signal modifiers. The processed signal should be connected to the “Ret” (Return) input of the macro. To avoid a constantly increasing volume in the feedback path the processing filters and signal modifiers should not amplify the signal beyond 1. For filtering the “Multimode - Accurate” macro of the classic modular library should be used (the “GainC” Knob should be set to 1).

## **Ladder Lowpass**

Lowpass filter modelled on the classic circuit patented by Bob Moog with smooth saturated resonance, self-oscillation at high resonance settings and frequency modulation (FM).

The filter calculates four different lowpass filters: a 1 pole, a 2 pole, a 3 pole and 4 pole filter. With each pole the damping of frequencies greater than the cut-off frequency increases with 6 dB per octave. For instance the 4-pole filter has a damping of 24 dB per octave. The “Poles” knob “scans” thru the output signals of the four filters. If the scanning position is in-between two filter signals the output signal is obtained by crossfading between these two.

## **Multimode - Accurate**

CPU friendly multimode filter with an accurate frequency response. This means that there are nearly no unwanted boosts or attenuations of frequencies. The “GainC” knob determines how much an increase of the resonance decreases the output level of the filter, which is especially helpful if the resonance is modulated. With the “GainC” knob set to 1 the amplification for all frequencies is smaller or equal 1 regardless of the resonance boost. This setting is recommended if the filter is inserted in a feedback path of a delay line (the “delay” and the “comb” macro allow external processing of the feedback signal) to prevent the level within the feedback loop from getting louder and louder.

The following filter types are available:

- 6 dB/oct low/high pass filter
- 12 dB/oct low/band/high pass filter
- 24 dB/oct low/band/high pass filter

### **Multimode - Resonance Limiter**

Multimode filter with a built-in resonance limiter. The “limit” control sets the threshold of this limiter in dB. If the bandpass filter signal exceeds this level the resonance for all filter types is reduced. This prevents loud filter responses at prominent frequency components of the filtered signal. The “F foll” control determines how much the threshold is decreased when the cut-off frequency is increased.

The “GainC” knob determines how much an increase of the feedback decreases the output level of the filter, which is especially helpful if the feedback is modulated.

The following filter types are available:

- 12 dB/oct low/band/high pass filter
- 24 dB/oct low/band/high pass filter

## **18.9. Delay**

### **Delay**

Delays the incoming signal by the time set with the “Delay” knob.

There are three “Unit” controls, which define the step size of the “Delay” knob. One of these controls is selected with the “Mode” switch. In “ms” mode the step size of the “Delay” knob is set in milliseconds. In “bpm” mode the delay time is set relative to the beats per minute (bpm) of the global clock of REAKTOR. The corresponding “Unit” control selects the note length. For instance “1/16” equals a 16th note. In “pos” mode the delay time is set relative to the measured time between incoming events at the “Pos” input of the macro. The step size of the “Delay” knob is the product of the measured time and the value of the “Unit” control. For instance, if the incoming position events have a 1/96 note resolution, the “Unit” control should be set to 6 to achieve a 16th notes step size. The “pos” mode should be selected if the delays should sync to a clock, which is independent from the global clock. Then the position events of this independent clock must be connected to the “pos” input of the macro.



The delay time can be modulated. The “MQ” control sets the quantization step size for the modulation signal in numbers of units set by the “unit” control. The amount of feedback can be set with the corresponding knob. The “Ex” switch activates the external feedback path. The delayed signal is sent out at the “Send” output of the macro so it can be processed by filters and other signal modifiers. The processed signal should be connected to the “Ret” (Return) input of the macro. To avoid a constantly increasing volume in the feedback path the processing filters and signal modifiers should not amplify the signal beyond 1. For filtering the “Multimode - Accurate” macro of the classic modular library should be used (the “GainC” Knob should be set to 1). The “Dry/Wet” knob defines the mix of the dry and the wet (delayed) signal.

## 18.10. Audio Modifier

All audio modifiers which distort the incoming signal like the “clipper” or “saturator” have a built in input amplifier to set the level of the incoming signal to zero dB. This is the case if the corresponding meter just doesn’t turn orange. It is recommended to do so since the value ranges of the control of the modifier are optimised for this signal volume.

Another common control is the gain correction “GainC” knob. It has the same functionality for all modifiers and is explained here for the “clipper”. If it is set to 0 the gain correction is turned off. That means that lowering the clipping level lowers also the output level. This is the right setting if the modifier is inserted in the feedback path of a delay since then the signal isn’t amplified. If set to 1 this loss in the volume is compensated. That means if the input signal has a level of 0 dB the output signal will also have a level of 0 dB independent of the set clipping level. This function is very helpful if the clipping level is modulated. Please note that this function only works properly if the incoming signal is levelled to 0 dB.

A lot of the modifiers have a symmetry (“Sym”) knob. This controls how much the positive and negative part of the signal is distorted differently. If it is zero the distortion is the same for both sides.

### Clipper

Clips the input signal at a controllable level.

## **Quantizer**

Quantises the amplitude of the incoming signal. The signal is distorted into a step waveform. Can be used to simulate low bit resolutions of vintage samplers.

## **Ringmodulator**

Signal modifier for amplitude and ring modulation. The “Mod Depth” control determines how much the amplitude of the input signal is modulated by the modulation signal. To achieve ring modulation this control has to be set to 1. Then the input signal is multiplied with the modulation signal.

## **Saturator**

Soft saturating overdrive modifier to simulate tube distortion and tape saturation effects.

## **Slew Limiter**

Slew rate limiter and smoothing filter.

The output signal follows the input signal with a limited rate. Different rate limits can be set for the rising signal and falling signal. This macro can be used to smooth modulation signals in general and to realise portamento (smooth gliding from one note pitch to the next).

## **Waveshaper**

Signal modifier that shapes the input signal using 2 adjustable breakpoints.

## **Wrapper**

Wraps or folds the incoming signal around an adjustable limit. Very powerful if used to shape oscillator signals. The results are similar to “Sync” or “PWM” (Pulsewidth-modulation) sounds.

## 18.11. Event Processing

### 0-1 to 0-127 Range Converter

Multiplies the incoming signal with 127. Should be used if a modulation signal should be connected to a “P” (Pitch) input of a module or macro, which expects a value between 0 and 127. Can also be used to transform a modulation signal into position events so it can address positions in a step sequencer.

The output can be quantised to integer steps.

### Quantizer

Distorts the incoming signal into a step waveform by rounding its values to the nearest multiple of the set step size.

### Randomizer

Randomises the incoming events, which means that a random number in a definable value range is added.



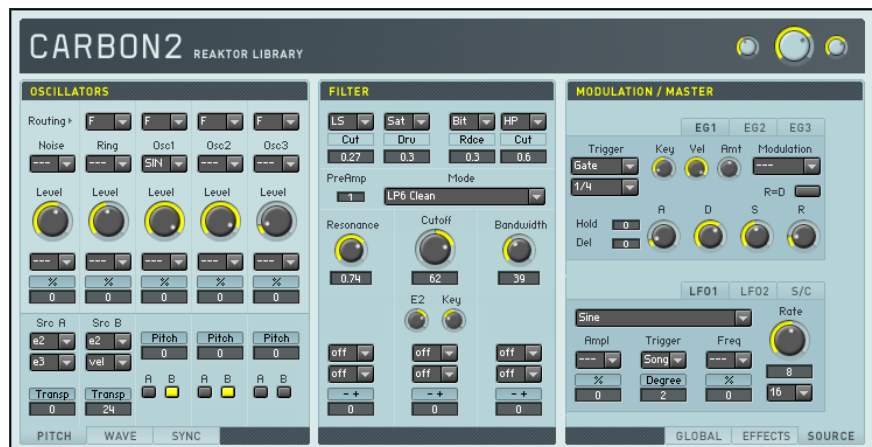


# **REAKTOR 5**

## **Instrument Guide**

# Synthesizer

## Carbon 2



Carbon 2 is based on REAKTOR 4's well-known workhorse synthesizer, but it has been completely rebuilt. In particular the oscillators and filters are now based on REAKTOR Core components developed particularly for this instrument. The panel has been optimized for usability, with a clear structure providing fast access to all parameters while hiding the technical complexity. Basically, Carbon 2 is a classical subtractive synthesizer. The signal of the three oscillator section (left column of the panel) passes through a multi-mode filter (middle column) and is then routed to the effect units (right column). Several modulation sources such as envelope generators and LFOs (placed in a second page in the right panel column) and the global parameters (a third page in the right column) control the sound, adding additional liveliness and movement.

### Oscillators

The oscillator section produces the instrument's basic signal. Three oscillator slots provide several different waveforms; along with traditional analogue types like sine and sawtooth there is a digital wavetable oscillator containing a wide array of waveforms that can be crossfaded smoothly. A noise generator and a ring modulator based on the signal of the three main oscillators are added for a total of five basic sound sources.

Each oscillator slot offers control over volume, pitch, and waveform synchronization. The pitch and sync controls are placed in two pages at the bottom of the panel, grouped with a third page controlling the waveform. This third page is only active if the digital wavetable or the doubled sawtooth is selected.

Main	Routing	Sets the destination of the respective oscillator's signal. On [F], the sound is sent to the [Filter] section; switching to [D] bypasses the filter and routes the signal directly to the effect units.
	Noise	Switches the white noise generator on or off.
	Ring	Selects which oscillator signals are fed into the ring modulator. Switch off to save CPU power if the ring modulator is not used.
	Osc1/2/3	Selects the waveform of each oscillator slot. Along with the standard waveforms (sawtooth, pulse, triangular, sine and noise), you will find a doubled sawtooth, a quantized sine, a buzz oscillator based on a noise generator, and a digital wavetable. (See the [Wave] page for additional information on the doubled sawtooth and the digital wavetable.)
	Level	Sets the slot's volume level.
	Level Modulation Source	Selects the slot's volume level modulation source.
	Level Modulation Amount	Sets the amount and polarity of modulation applied to the slot's volume level. Clicking on the control's title bar restores the value to its default.
Pitch	A/B Modulation Source	Selects sources to modulate the oscillators' pitch. The two individual slots ([A] and [B]) can mix up to two sources.
	A/B Modulation Amount	Adjusts the amount and polarity of modulation applied to the oscillators' pitch. The left side of the control adjusts coarse values, the right side is used for fine-tuning; clicking the control's title bar restores the value to its default.
	Osc1/2/3 Pitch Shift	Transposes the oscillators' sound respectively. The left side of the control adjusts coarse values, the right side is used for fine-tuning; by clicking on the controls title bar with the mouse the value is reset to its default.
	Osc1/2/3 Modulation Switch A/B	Turns modulation of the oscillator's pitch by modulation slot [A] or [B] on or off.

Wave	A/B Modulation Source	Selects sources to modulate the waveform. The two individual slots ( [A] and [B] ) can mix up to two sources. This will show no effect until the doubled sawtooth or the wavetable is selected in [Osc1/2/3].
	A/B Modulation Amount	Adjusts the amount and polarity of modulation applied to the waveform. Clicking on the control's title bar resets the value to its default. This will show no effect until the doubled sawtooth or the wavetable is selected in [Osc1/2/3].
	Osc1/2/3 Waveform Control	This either selects a digital waveform from the wavetable, or – if the doubled sawtooth is activated in [Osc1/2/3] – this controls the ratio between the phases of both sawtooth waves.
	Osc1/2/3 Modulation Switch A/B	Turns modulation of the waveform selection by modulation slot [A] or [B] on or off.
Sync	Gate Sync Switch	Turns synchronization of the oscillators' waveforms to the MIDI gate on or off. If on, all three oscillators are reset to the phase adjusted in [Gate Sync Phase] when a note is pressed.
	Gate Sync Phase	Controls the phase to which all oscillators are set on MIDI gate events. Clicking on the control's title bar restores the default value.
	Osc2/3 Sync Switch	Switches on or off the synchronization of the oscillators 2 and 3 respectively to the signal of oscillator 1. If on, the oscillator is reset to the phase adjusted in [Osc2/3 Sync Phase] when the signal of oscillator 1 rises above zero. (See also [Osc2/3 Mode Fade].)
	Osc2/3 Sync Phase	Controls the phase to which the oscillators 2 and 3 are reset when the signal of oscillator 1 rises above zero. Clicking on the control's title bar restores the default value. (See also [Osc2/3 Mode Fade].)
	Osc2/3 Mode Fade	Interpolates between hard synchronization (at low values) and soft synchronization (at high values). In hard synchronization mode the oscillator is always reset if the signal of oscillator 1 rises above zero; with soft synchronization this is not always the case, producing a mix between the synced waveform and the non-synced one. Clicking on the control's title bar restores the default value.



## Filter

The filter section is placed between the oscillators and the effects; it sculpts the oscillators' basic sounds. Before the signal is routed to the filter it passes two effects that provide saturation and quantization, as well as additional low- and high-shelf equalizers. The filter itself contains several modes, optimized for a warm yet crisp sound. You'll find standard low-pass and high-pass, band-pass, and band-reject filters, a special feedback filter (called [Zwnl]), and a peak EQ and comb filter. After the main filter comes another effect section, similar to the previous one.

Pre-Filter Effects	Effect A/B Mode Select	Selects the effect units applied to the signal before it passes to the filter. There are low and high shelf EQs in the left [A] menu and saturation and quantization in the subsequent right [B] menu.
	Effect A/B Amount	Sets the parameter of the effect unit selected by [Effect A/B Mode Select]. For the equalizers, this is the amount of damp or boost applied to the signal. For the saturator it's the amount of drive, and for the quantizer it's the amount of distortion.
Main	PreAmp	Controls the level correction of the signal after it has passed the [Pre-Filter Effects] section and before it enters the main filter.
	Mode	Selects the filter mode. There are high-pass, bandpass and band-reject filters, several low-pass modes, a feedback lowpass, a peak equalizer, and a comb filter.
	Cutoff	Sets the frequency of the filter.
	Resonance	Sets the resonance of the filter.
	Bandwidth	Sets the width of the band for the bandpass and bandreject filters. If the peak equalizer is selected, this parameter sets the amount of boost applied.
	E2	Controls the amount and polarity of modulation applied to the cutoff control by the second envelope generator. Turn to the left for negative modulation, i.e. low cutoff values at high envelope signals. Turn to the right for normal positive modulation.
	Key	Controls the amount and polarity of modulation applied to the cutoff control by the current pitch. Turn to the left for negative modulation, i.e. low cutoff values at high pitches. Turn to the right for normal positive modulation. This modulation is independent of the Key Scaler of the [Modulation] section.

	Cutoff/ Resonance/ Bandwidth Modulation Source	Selects the sources used to modulate the filter's cutoff, resonance and bandwidth. Up to two sources can be selected, and their signals are summed together. In case of the cutoff modulation, these signals are added to the hard-wired modulation by the second envelope generator and the MIDI pitch.
	Cutoff/ Resonance/ Bandwidth Modulation Amount	Adjusts the amount and polarity of modulation applied to the filter's cutoff, resonance and bandwidth. Clicking on the control's title bar restores the default value. In case of the cut-off modulation, this amount doesn't affect the hard-wired modulation by the second envelope generator and the MIDI pitch.
	Effect A/B Mode Select	Selects the effect units applied to the signal after the filter, before it gets routed to the main effect units. You'll find saturation and quantization in the left [A] menu, and lowpass and highpass filters in the right [B] menu.
	Effect A/B Amount	Sets the parameter of the effect unit selected by [Effect A/B Mode Select]. For the saturator this is the amount of drive; for the quantizer the amount of distortion; and for both filters the cut-off frequency.
	Post- Filter Effects	

## Effects

The effects additionally enhance the instrument's sound. There are five units: a pitch shifter, a phaser, a chorus, an equalizer and a delay. These standard effects are engineered for the finest of results.

	Power & Mix	Each effect unit provides a power switch and a mix button. The mix button crossfades between the dry, unprocessed signal (at the left) and the wet effect sound (at the right). To save CPU power, turn the power switch off when the specific effect is not in use.
Pitch Shifter	Shift L / R	Determine the pitch shift of the left and right channel respectively in semitones.
	Grain Size L/ R	Adjust the grain size of the pitch shifting algorithm for the left and right channel respectively. Turn to the left for large chunks and echoic sounds, to the right for tiny grains and an accurate pitch shift.
	Feedback Reverse	Controls the amount of feedback. Switches between forward and reverse grain playback.

Phaser	Center Frequency	Sets the center frequency of the filters that produce the phaser signal.
	Modulation Rate	Sets the speed at which the [Center Frequency] is modulated.
	Phase	Sets the phase of the LFO modulating the [Center Frequency]. (See also [Modulation Rate].)
	Depth	Sets the amount of modulation.
	Resonance	Sets the resonance of the internal filters.
	Feedback	Sets the amount of feedback.
Chorus	Delay	Sets the main delay of the chorus.
	Depth	Sets the amount of modulation applied to the [Delay] time.
	Rate	Sets the speed at which the [Delay] time is modulated.
Equalizer	Bass Boost	Controls the boost (or damping) applied to the bass frequencies below 300 Hz.
	Mid Frequency	Adjusts the frequency of the peak equalizer applied to the middle frequency spectrum.
	Mid Boost	Controls the boost (or damping) applied to the middle frequencies around the [Mid Frequency].
	Mid Resonance	Sets the resonance of the mid equalizer.
	High Frequency	Adjusts the frequency of the high shelf equalizer.
	High Boost	Controls the boost (or damping) applied to the frequencies above the [High Frequency].
Delay	Delay L / R	Sets the delay times of the left and right channel respectively. The time is controlled in increments selected by the [Quantize] control.
	Fine L / R	Adds an offset to the values controlled by [Delay L / R] in milliseconds.
	Quantize	Selects the unit by which the delay times are quantized. Sixteenth notes and eighth triplets are available.
	Feedback	Sets the amount of feedback.
	Wrap	Controls the amount of cross-feedback. Turn to the left to route the every channel's feedback to itself; turn to the right to route it to the other channel.
	Resonance	Sets the amount of resonance applied to the low-pass and high-pass filters within the feedback circuit.
	Lowpass	Controls the frequency of the low-pass filter within the feedback circuit.
	Highpass	Controls the frequency of the high-pass filter within the feedback circuit.

## Modulation Sources

Several modulation sources are available: two ADSR envelope generators, a recordable envelope, and two LFOs combined with a key-scaler that provides four independent control points and four freely assignable MIDI controllers. The envelope generators and LFOs offer several types of MIDI clock interaction for rhythm-based modulation effects.

Envelope Generators 1/2	Trigger	Selects the events that re-trigger the envelope generator. [Gate] only activates the MIDI gate signal. [Clock Gate] re-triggers the envelope at each unit selected by [Quantization] as long as the MIDI gate is open. [SP Clock Gate] is similar, but synchronizes the quantization to the global MIDI song position; therefore, the MIDI clock has to be running. (See also [Globals][EG Mode].)
	Quantization	Selects the metrical unit used to re-trigger the envelope if [Trigger] is set to [Clock Gate] or [SP Clock Gate].
	Key	Controls the amount and polarity of modulation applied to the envelope's transition times by the current pitch. Turn to the left for negative modulation, i.e. shorter attack, decay and release times at low pitches. Turn to the right for normal positive modulation, i.e. longer times at low pitches.
	Velocity	Controls the current velocity's influence on the envelope amplitude. At low values the envelope triggers with the same amplitude; at high values the MIDI velocity determines its peak value.
	Transition Time Modulation Select	Selects the additional modulation applied to the envelope generator's transition times. The attack phase can be modulated by the MIDI velocity while the decay time can be modulated by the velocity and the four MIDI controllers (see [MIDI Controllers]). The amount and polarity of modulation is controlled by [Transition Time Modulation Amount].
	Transition Time Modulation Amount	Controls the amount and polarity of modulation applied to the destination selected by [Transition Time Modulations Select]. Turn to the left for negative modulation, i.e. shorter attack or decay times at low modulation source values. Turn to the right for normal positive modulation, i.e. longer times at low values.
	Attack	Sets the attack time of the envelope generator.
	Decay	Sets the decay time of the envelope generator.
	Sustain	Sets the sustain level of the envelope generator.

Envelope Generator 3	Release	Sets the release time of the envelope generator.
	Hold	Sets the duration of an additional hold phase between attack and decay.
	Delay	Adds an initial delay period before the trigger signal restarts the envelope
	R=D	Links the release time to the decay time. If on, the value adjusted by [Decay] is also used to control the release phase.
	Record	Arms the recordable envelope. The recording is started when a MIDI gate is received and ends when the gate closes. All movements of the [Value] knob are stored and can be played back as envelope (see [Play]).
	Play	Enables playback of the recorded movements, triggered like an envelope by MIDI gate signals.
	Loop	Loops the recorded movement on playback.
LFO 1/2	Value	When recording (see [Record]), every movement of this knob is stored to the memory. During playback (see [Play]), the knob displays the recorded movements.
	Waveform	Selects the waveform of the Low Frequency Oscillator. There are the standard waveforms [Sine], [Triangular], [Pulse], and [Random Steps], and several derivations: [Pulse+] is a pulse waveform with all negative values clipped to 0; [Saw Up+] and [Saw Down+] are triangular forms with only rising resp.falling ramp and only positive values; [Hsin+] is a multiplication of [Pulse+] and [Sine] etc.
	Amplitude Modulation Source	Selects the source used to modulate the LFO's amplitude. Clicking the control's title bar restores the value to its default.
	Amplitude Modulation Amount	Adjusts the amount and polarity of modulation applied to the LFO's amplitude.
	Trigger Mode	Selects the events that re-trigger the LFO. In [Freerun] mode no reset occurs; in [Gate] mode the LFO is set to the phase adjusted by [Reset Phase] on a MIDI gate event. [Clock Gate] is similar to [Gate] mode but also activates a grid for the LFO's frequency (see [Rate]). [SP Clock Gate] additionally synchronizes the reset to the global MIDI song position.
	Reset Phase	Adjust the phase to which the LFO is set on re-triggering events.

	Rate Modulation Source	Selects the source used to modulate the LFO's frequency. If [Trigger Mode] is set to [Clock Gate] or [SP Clock Gate], frequency modulation is not available.
	Rate Modulation Amount	Adjusts the amount and polarity of modulation applied to the LFO's frequency. Clicking the control's title bar restores the value to its default. If [Trigger Mode] is set to [Clock Gate] or [SP Clock Gate], frequency modulation is not available.
	Rate	Sets the frequency of the LFO. If [Trigger Mode] is set to [Clock Gate] or [SP Clock Gate], a grid is applied to this control, quantizing the LFO's rate to the metrical units selected in [Rate Quantization].
	Rate Quantization	Selects the metrical unit used as quantization grid for [Rate] when [Trigger Mode] is set to [Clock Gate] or [SP Clock Gate].
KeyScaler	Sliders	Provides a signal derived from the current pitch that can be used as modulation source. The four sliders define the function used to map the MIDI pitch onto the modulation signal: At low pitches, the value of the leftmost slider is used as modulation signal; at high pitches the value of the rightmost slider is selected. In between, interpolation occurs, using the two middle sliders as control points. In addition to the normal signal, there is a modulation source that multiplies the key-scalers value by the current MIDI velocity.
MIDI Controllers	Faders	The leftmost fader is hard-wired to the MIDI modulation wheel. All others can easily be assigned to any MIDI Continuous Controller via MIDI Learn. They are available as modulation sources, named C1, x1, x2 and x3.

## Global Controls

The global controls access several different functions. First – and most important – the voice allocation of the synthesizer can be controlled, providing polyphonic and monophonic modes; by selecting the unison mode all available voices are set to the same pitch (as in a monophonic synth), but each one is slightly detuned. This results in waveform interference and a thick, chorus-like sound. Monophonic modes also provide portamento.

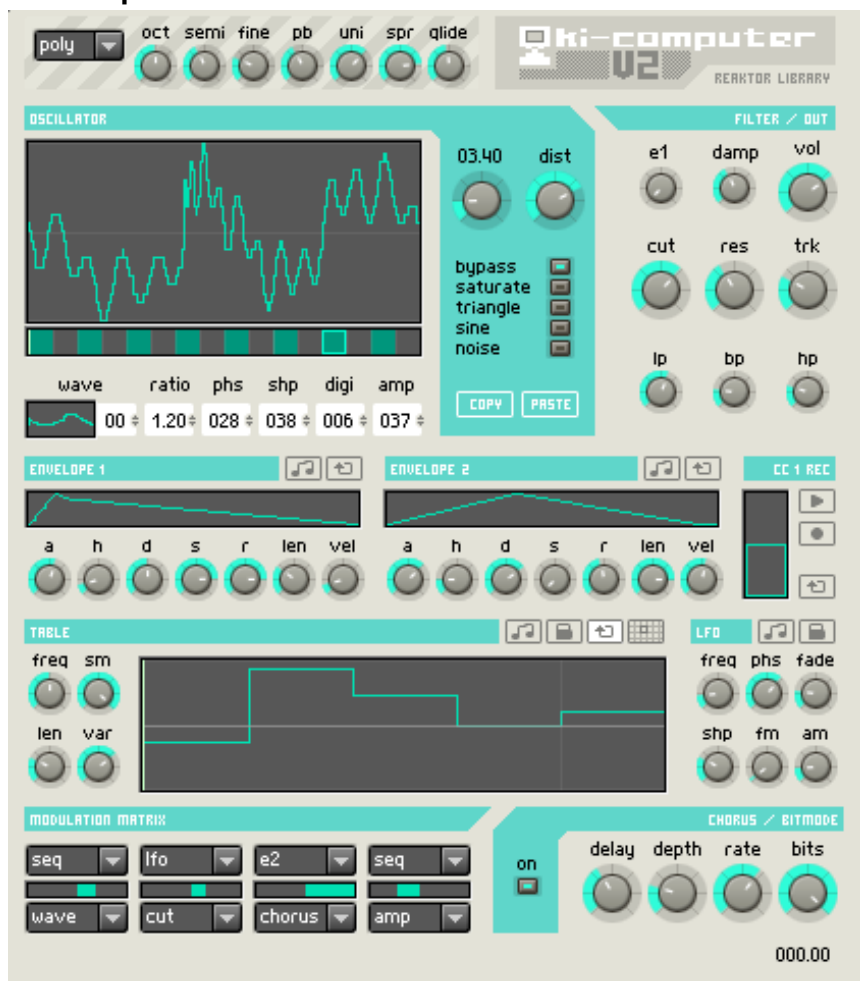
Parameters determine the master pitch shift and MIDI pitchbend range, and adjust global tremolo and vibrato. Voices' position within the stereo field can also be adjusted.

Gate Mode	Selects the global operation mode. [Poly] selects the only polyphonic mode; portamento does not work in this mode (see [Glide Speed]). [Mono] results in a monophonic gate signal that is triggered on every MIDI note. [Legato] is similar but generates a new gate trigger signal only when the gate has been closed before, i.e. no note is already pressed. [Uni Mono] and [Uni Legato] activate the unison modes: A monophonic gate signal is used for all voices, but all available voices are used and detuned by the [Unisono] and [Unisemi] controls.
Envelope Mode	Selects the envelopes' behavior during the release period if a new attack is triggered. [Re-trigger] starts the attack phase beginning with the current envelope amplitude; [Reset] starts the attack with a value of zero. Thus, [Reset] might lead to unwanted clicks if used without care.
Unisono	Sets the amount of detuning applied to each voice when [Uni Mono] or [Uni Legato] is selected as [Gate Mode]. Slight detuning results in thick, chorus-like sounds.
Unisemi	Sets the amount of pitch shifting applied to each voice when [Uni Mono] or [Uni Legato] is selected as [Gate Mode]. This acts like the [Unisono] control but detunes the voices in semitones, e. g. a value of 12 will set all voices one octave apart.
Drift	Enables a drift mode that slightly detunes higher pitches. This results in a more analogue like sound.
Key	Activates key-scaling for the unisono control. If pressed, the [Unisono] value is lowered automatically at high pitches for a more constant sound over the complete pitch range of the instrument.
Velocity	Selects the mapping applied to the MIDI velocity. While [Linear] doesn't change the velocity, [Log] results in a compressor like effect while [Expo] produces the opposite effect.
Coarse	Sets the global tuning of the instrument in semitones, ranging from -63 to +64.

Fine	Sets the global tuning of the instrument in semitones, ranging from -0.5 to +0.5
Glide Speed	Adjusts the speed at which new pitches are reached if they are slurred, i.e. if the previous note was still held when the new one was pressed. This portamento effect only works in monophonic modes (see [gate Mode]).
Pitchbend Range	Sets the range in semitones by which the MIDI pitchbend wheel transposes the global pitch.
Vibrato Mode	Selects whether vibrato is off, on, or faded in by the MIDI modulation wheel.
Vibrato Amount	Sets the amount of vibrato. Clicking the control's title bar restores the value to its default.
Vibrato Style	Selects between three different vibrato modes.
Key	Adjusts the amount of key scaling applied to the vibrato. Turn to the left for no scaling, to the right for less vibrato at low pitches, producing a more musical effect.
Tremolo Mode	Selects whether tremolo is off, on, or faded in by the MIDI modulation wheel.
Tremolo Amount	Sets the amount of tremolo. Clicking the control's title bar restores the value to its default.
Vibrato & Tremolo Frequency	Sets the speed of both vibrato and tremolo.
Voice Panning Switch	Selects whether the instrument's voices are placed at different positions within the stereo field. Especially in combination with the [Unisono] control this can produce impressing spatial effects.
Voice Panning Amount	Sets the amount of voice panning. Clicking the control's title bar restores the value to its default.
Master 1/2	Defines the instrument's output level. Use the large middle knob to adjust the preset's maximum level; the smaller knob to the right controls the instrument's output amplitude in all patches.
Key Amp	Adjusts the amount of automated amplitude correction in respect to the synthesizer's pitch. Turn to the left for no influence of the pitch onto the output level, to the right to damp high pitches. This can be used to simulate the sound of analogue synthesizers.



## Oki Computer 2



If you get excited by words such as analog and vintage, look away now. Oki Computer 2 is a compact wavetable synthesizer, a specialist in digital lo-fi sounds that hails back to the era of 8-bit beeps and bleeps... It is also rather capable at creating buzzing leads, rhythmic sequences, and odd tasty bass tones.

Oki Computer 2's panel is compact but packed with features. Thankfully, most sections should be fairly straightforward to the average synth user. However, the [Oscillator] section is somewhat unique and users are strongly encoura-

ged to read this part of the manual. Oki Computer 2 features a bank of 50 waves. For every patch you can load any 16 of these waves into the oscillator in any order. This flexibility represents a major improvement over the original ensemble (where the oscillator was permanently ‘hardwired’ to the same 16 waves). What’s more, each wave loaded into the oscillator can be processed in a variety of ways.

## MIDI In

The drop-down list at the panel’s top-left switches between polyphonic and monophonic operation modes. In polyphonic mode, Oki Computer operates as a standard poly-synth. Monophonic mode does not restrict the number of voices to 1; it enables some very musical features – legato, glide, and unison.

Gate Mode	Selects whether the instrument is used as polyphonic or monophonic synthesizer.
Unison	Determines the number of simultaneous voices. This is only active if [Gate Mode] is set to [mono].
Spread	Defines the amount of voice detuning in semitones. This is only active if [Gate Mode] is set to [mono].
Glide	Sets the amount of portamento, i.e. the time used to reach a new MIDI pitch. This is only active if [Gate Mode] is set to [mono].
Octave	Transposes the pitch of the entire oscillator in octave steps.
Semitone	Transposes the pitch of the entire oscillator in semitone steps.
Fine	Fine-tunes the entire oscillator’s pitch.
Pitchbend	Defines the MIDI pitch bend wheel range in semitones.

## Oscillator

The [Wavetable Position Bar], located beneath the main oscillator window, is perhaps the most difficult part of the synthesizer to understand. This bar has two purposes. Firstly, the square box indicates the current wave slot selected for editing (there are 16 slots). Secondly, the bright green line indicates the current wavetable position. The current wavetable position is set by the [Wavetable Position Knob] (to the left of the drive knob), plus any modulation routed to the wave table position (see [Modulation Matrix]).

The best way to explain how the [Wavetable Position Bar] works is by example: Click on the snapshot menu and recall preset number 1 - ‘Default’. In this preset, the oscillator is loaded with 16 sine waves (needless to say, this doesn’t sound particularly interesting). Click the leftmost box on the

Wavetable Position Bar - this will select the first slot for editing. Note that in the box labeled [Wave] (beneath the [Wavetable Position Bar]) a picture of a sine wave is displayed, with a number zero next to it. This indicates that the sine wave (wave number zero from the master bank) is loaded into the current slot. To load a different wave, click and drag vertically on the Wave Selector. Now click on the second wave slot (i.e. the adjacent dark gray box). The Wave Selector will display a sine wave (remember this patch had 16 identical sine waves loaded). Now try loading a different wave to slot 2, again by clicking and dragging on the Wave Selector.

In the 'Default' snapshot, the [Wavetable Position Knob] is set to 1.00. This means that when you play a note, you will hear (and see) the wave loaded into slot 1. Press a note on your keyboard, and slowly move the knob from 1.00 to 2.00. You will hear and see the wave loaded into slot 1 morph into the wave loaded into slot 2. Notice how the wave position indicator moves correspondingly. This is the key to how Oki Computer 2 produces dynamic sounds: by morphing between adjacent waves in the wavetable. While this can be done manually with the wave position knob, things get much more interesting when the various modulators (e.g. envelope, sequencer, LFO) are used to mix between waves.

Apart from the Wave Selector, all the controls underneath the wavetable position bar are used to modify wave shape. When using these controls, it is important to remember that they only affect the wave in the selected slot (i.e. the green box), which is not necessarily the wave shape currently being played (i.e. the green line).

Ratio	Sets the number of times the wave shape will repeat in a single oscillator cycle. Note that the integer and decimal values can be set independently, also note that adjusting the ratio will cause pitch shifting.
Phase	Rotates the wave start position within the oscillator cycle.
Shape	Skews the wave shape to either the left or right (on the pulse wave this is identical to a pulse width control).
Digitize	Reduces the wave's bit depth.
Amp	Attenuates the wave volume.
Copy	Stores the current settings into an edit buffer that can be read out again by the [Paste] button.
Paste	Recalls the data from the edit buffer (see [Copy]).
Distortion Amount	Controls the amount of distortion. (See also [Distortion Mode].)

Distortion Mode	Selects the way the signal is distorted. [Saturate] applies a 'standard' saturation curve to the signal. [Triangle] and [Sine] both involve wrapping their respective shapes around the input signal. When used on a sine wave, these two functions can sound somewhat reminiscent of FM. [Noise] enables a noise generator.
-----------------	--

## Filter / Out

This section controls the shaping applied to the sound's frequency spectrum (filter) and amplitude.

Amplitude Envelope Mode	Selects the main output envelope. In most cases, [E1] will be the preferred choice. Sometimes however, you may want to use Envelope 1 for modulation purposes only. In this case, select either [G] (MIDI gate, ignoring velocity) or [Vel] (MIDI gate, including velocity).
Damp	Controls the amount of high-frequency damping.
Volume	Sets the main output volume in decibels.
Cut-off	Sets the filter's cut-off frequency.
Resonance	Adjusts the filter's resonance amount.
Track	Defines the amount of cut-off pitch tracking. At 100%, cut-off is increased by one semitone for each increment in MIDI pitch. At -100%, cut-off is decreased by one semitone for each increment in MIDI pitch. At +/- 200%, cut-off changes by two semitones for every one semitone change in pitch.
Low-pass, Band-pass, High-pass	Determines the mix ratio of the high-pass, band-pass and low-pass components of the filter output signal.

## Envelope, CC1, Sequencer and LFO

Oki Computer 2 features two envelope generators. Both can be used for general modulation via the Modulation Matrix, but envelope 1 can also be routed directly to output volume in the [Filter / Out] section. Otherwise, the envelope generators are identical.

The CC1 section allows you to record modulation wheel movements. To record a sequence, click the [Record] button. The button will flash, indicating that it is armed and waiting. Recording will begin when a MIDI note is pressed, and finish when the note is released (or when all recording memory is used up). You can record movements with the mouse, or the CC1 knob on a MIDI controller. As long as the [Play] button is depressed, recordings will play back each time a note is triggered. The recording is sent to MIDI CC1. Therefore, to use the recording as a modulation source, select CC1 in the Modulation Matrix. Note that even though sequences are recorded monophonically, playback operates in full polyphony.

The sequencer is a highly flexible modulation source. It can operate as an arpeggiator, a custom shape LFO or an additional envelope. You can draw its steps with the mouse.

You'll find a standard LFO next to the sequencer.

Envelope	Attack	Controls the attack time of the envelope generator.
	1/2	
	Hold	Controls the hold time of the envelope generator.
	Decay	Controls the decay time of the envelope generator.
	Sustain	Controls the sustain level of the envelope generator.
	Release	Controls the release time of the envelope generator.
	Speed	Multiplies the overall envelope time.
	Velocity	Determines the extent to which envelope amplitude is linked to velocity.
CC1	Clock Sync	Synchronizes the envelope times to the global MIDI tempo.
	Loop	Activating this button results in the attack, hold and decay stages looping when MIDI notes are depressed.
	Record	Arms the CC1 recorder.
	Play	Enables playback of the recorded movements, triggered like an envelope by MIDI gate signals.
	Loop	Loops the recorded movement on playback.

Sequencer	Clock Sync	Synchronize the sequencer to the MIDI clock. Note that when both [Clock Sync] and [Phase Lock] are activated, the sequencer becomes locked to the MIDI song position.
	Phase Lock	Locks the phase of the sequencer. When this is active, MIDI note events will not re-trigger the sequencer. Note that when both [Clock Sync] and [Phase Lock] are activated, the sequencer becomes locked to the MIDI song position.
	Loop	When enabled, the sequencer will loop indefinitely, otherwise it will play back only once when triggered.
	Snap	Activates a vertical grid with a step size of 1/12 of the complete height.
	Frequency	Determines the sequencer speed.
	Length	Sets the sequencer length in steps.
	Smooth	Determines the amount of interpolation between adjacent steps (at hard-right, the sequencer produces smooth envelope-type output).
	Variation	Imparts a kind of swing onto the sequencer movement, so steps play alternately faster and slower. Center this control for equal length steps.
LFO	Clock Sync	Synchronize the LFO to the MIDI clock. Note that when both [Clock Sync] and [Phase Lock] are activated, the LFO becomes locked to the MIDI song position.
	Phase Lock	Locks the phase of the LFO. When this is active, MIDI note events will not re-trigger the LFO. Note that when both [Clock Sync] and [Phase Lock] are activated, the LFO becomes locked to the MIDI song position.
	Frequency	Sets the LFO speed.
	Phase	Determines the point in the LFO wave where oscillation begins when a note is triggered. This only function when [Phase Lock] is off.
	Fade	Sets the fade-in time of the LFO (i.e. the time taken to reach full amplitude).
	Shape	Skews the LFO shape to either the left or right.
	FM / AM	Determine the amount which the modulation wheel (including recorded movements) modulatea frequency and amplitude respectively.

## Modulation Matrix

The modulation matrix enables any four modulation sources to be routed to any four destinations. You can select modulation sources using the upper drop-down menus. Destinations are selected with the lower menus. The sliders in-between these menu set the modulation amount. The full list of modulation sources and destinations is summarized in the following table:

Sources	Vel	MIDI note on velocity	(0 to 1)
	PB	MIDI pitchbend wheel	(-1 to 1)
	CC1	MIDI CC1 - the modulation wheel. Note that recorded CC1 movements (in the recordable envelope section) are routed to this parameter.	(0 to 1)
	E2	Envelope generator 2	(0 to 1)
Destinations	Seq	The sequencer	(-1 to 1)
	LFO	The LFO	(-1 to 1)
	Amp	Output volume	(-100% to +100%)
	Pitch	Oscillator pitch	(-12 to +12 semitones)
	Wave	Oscillator wave position	(-16 to +16)
	Cutoff	Filter cutoff	(-60 to +60 semitones)
	Chorus	Chorus frequency	(-100% to 100%)

## SteamPipe 2



SteamPipe 2 is a physical-modeling synthesizer that effectively models air being blown through a tunable pipe. It uses a tuned resonator to create bowed, blown, and plucked sounds, as well as strange new hybrid sounds. In addition to a tuned all-pass filter and many controls for the “shape” of the pipe, there is a mod wheel-controlled filter to achieve damping and breath noise effects. The excellent-sounding SpaceMaster Deluxe reverb unit adds dimension to the overall signal. You can find it on panel B.

SteamPipe 2 simulates air passing through a pipe of variable size and resonance. It's physical-modeling techniques use contoured noise signals passing through tuned and filtered feedback delays. The ensemble is basically split into two parts: Steam and Pipe. The Steam module generates shaped and filtered noise. Think of the Steam module as SteamPipe 2's oscillator. Steam provides the sound energy that will be pitch-formed by the Pipe. The Pipe module gives the “wind” pitch and resonance. The patch also has an ADSR volume envelope and a low pass filter. Both can be modulated by key- and velocity-tracking.

Steam Pipe 2 can be a very expressive synthesizer, so make sure that you plug in your MIDI keyboard and check out the presets with the mod wheel in action.

### Steam

The timbral shaping of the DC/Noise source occurs in the Steam section. The low pass filter works in 1-pole or 2-pole mode, though the resonance control only applies to the 2-pole filter. After the noise is filtered, the signal is fed into the Pipe module.



Envelope	Attack	Sets the attack time of an ADSR envelope triggered by MIDI gate events and used to generate a short initial steam signal; logarithmic control.
	Decay	Sets the decay time of an ADSR envelope triggered by MIDI gate events and used to generate a short initial steam signal; logarithmic control.
	Sustain	Sets the maximum level the envelope will reach. This gets modulated by velocity if [VelSns] is on.
	Release	Sets the time that passes until the envelope is completely faded out after the note-off signal.
	Velocity	Controls the velocity sensitivity of the envelope. The higher this value is, the higher the peak value of the envelope will be.
	Scaling	This scales the envelope times depending on the pitch of the incoming MIDI notes. Turn to the left for no keyboard scaling, to the right for shorter envelope times on higher notes.
	Legato	Toggles legato mode on or off. If on, the envelope restarts only when the gate changes from zero to positive.
Generator	DC / Noise	Crossfades between the DC component at the left and filtered noise at the right. The mixed signal is used as steam input of the resonating pipe.
	Cutoff	The cutoff frequency of the low-pass filter.
	Reso	Sets the level of resonance of the filter. Only works when the filter is in 2-pole mode.
	Poles	Toggles between 1-pole and 2-pole low-pass.
	Key-track	Controls the key-tracking of the filter. This will scale the cutoff frequency according to keyboard position. The lower the note pitch, the lower the cutoff frequency will be.
	Vel-Track	Controls the velocity scaling of the filter. Turn to the right for higher cutoff frequencies at higher MIDI velocities.
	Env-Amt	Sets the amount of envelope to the cutoff frequency.

## Pipe

The Pipe module is made up of a number of sub-modules for creating pitch and resonance. The noise signal is fed from a single tuned delay providing pitch, into the [Allpass] module for generating resonance. Next, a [Saturator] receives the signal and applies edge and break-up. The [MW Filter] completes the signal chain with an overall tone shaping stage. The [Feedback] and [Push-Pull] sections act on signals diverted from the main signal chain and passed back into it via feedback loops. Unlike the [Feedback] section, which simulates the pipe itself, the Push-Pull section controls the air and its oscillations within the pipe.

The [Delay Tune] module contains the tuned delay that provides pitch to the Steam. The [Tune] and [Fine tune] knobs allow you to set the signal's fundamental pitch. The A440 oscillator at the bottom of the ensemble provides a reference pitch for tuning purposes. The Delay pitch can be swept negatively or positively by the mod wheel, with the amount of modulation set by the [MW] knob.

The Allpass filter receives the tuned signal from the resonant delay. It can be turned on and off with the [Power] button in the [Allpass] section. This allpass can be tuned to create resonance effects. You can produce glassy, metallic and bell-like sounds by detuning the allpass filter against the delay. By adjusting the [Diffusion] knob, you can also create a variety of reverb sounds - the simulation of air echoing along a pipe's hard surface.

The Saturation module morphs between saturation and clipping, overdriving or breaking up the signal before it hits the MW Filter.

The [MW Filter], controlled by the mod wheel, features a 1-pole high pass followed by a 1-pole low pass filter. Each filter allows you to set a wheel-down and a wheel-up setting, making it possible to set up complex timbre changes and damping effects. Each filter can have its own [key track] setting.

The [Polarity] switch inverts the pipe polarity, changing the timbre of the sound. This often transforms high frequency tones to deep ones and vice versa.

The [Feedback] module processes the feedback in the signal chain. The [Rev-Time] knob extends or shortens the reverb generated by this feedback signal. The reverb signal can be muffled with the [Damp] control. Damping can be modified by [Key-Track] amount. High [Key-Track] values result in more damping on higher pitches. This allows SteamPipe 2 to emulate struck or plucked instruments like pianos, harps, and acoustic guitars.

Delay Tune	Tune	Sets the fundamental pitch of the signal. For standard musical tunings set it to the [A440] oscillator at the bottom of the patch.
	Fine Tune	Fine control of signal pitch.
	SREC	Sample rate error correction. Adjusts the tuning correction of the pipe. When the signal modifications in this patch and REAKTOR's sampling rate interfere with the physical model of SteamPipe, this extra-fine pitch tuning becomes necessary. Tune against the A440 section.
Feedback	Mod-Whl	Sets the amount of pitch modification by the MIDI modulation wheel. This simulates pitch changes of pipes getting blown softer or harder.
	Rev-Time	Adjusts the time of the pipe's reverberation, i.e. the amount of damping applied to the feedback's signal before it is mixed again with the new incoming signal. The longer the reverb time, the more the incoming noise steam signal becomes a tone with recognizable pitch.
	Damp	Sets the amount of high frequency damping of the pipe at key-up.
	Key-track	Controls the feedback's key-tracking. Turn to the right for longer reverb times at high MIDI pitches.
Allpass Tune	Tune	Controls the pitch of the allpass resonance. If the allpass filter is switched off, this control shows no effect.
	Fine Tune	Fine tunes the allpass resonance pitch. If the allpass filter is switched off, this control shows no effect.
	SREC	Sample rate error correction. Adjusts the tuning correction of the pipe. When the signal modifications in this patch and REAKTOR's sampling rate interfere with the physical model of SteamPipe, this extra fine-tuning of pitch becomes necessary. Tune against the A440 section. If the allpass filter is switched off, this control shows no effect.
	Mod-Whl	Sets the amount of pitch modification by the MIDI modulation wheel. This simulates pitch changes of pipes getting blown softer or harder.

Allpass	On / Off	Turns the allpass module on or off. Switch on for additional attack effects of the pipe's sound.
	Diffusion	Sets the diffusion of the resonances generated by the allpass module. Turn to the left for additional attack effects of the pipe's sound. It also enhances the sound of harmonic frequencies which are not multiples of the main pitch, like e.g. in bells.
	Offset	Sets the offset amount added to the reverberating steam signal. This parameter influences the incoming steam and its reverberation in the pipe. It interacts tonally with the Polarity button.
Push-Pull	Push	Sets the amount of reverberating steam.
Saturation	Soft / Hard	Controls the balance between soft saturation and hard clipping.
	Symmetry	This parameter introduces level asymmetry into the signal. With increasing asymmetry the positive part of the signal is reduced.
Polarity	Polarity	This control inverts the polarity of the pipe, thereby changing the timbre of the sound. It interacts tonally with the [Push-Pull] section.

Mod-Wheel to filter	Hi Pass 0	Sets the cutoff frequency of an additional highpass filter within the pipe to enhance formant frequencies. The formants are modified by the pressure in the pipe (not by the pipe's pitch). The pressure can be controlled by the mod wheel. At low modulation wheel values this knob is used to determine the formant frequency.
	Hi Pass 1	Sets the cutoff frequency of an additional highpass filter within the pipe to enhance formant frequencies. The formants are modified by the pressure in the pipe (not by the pipe's pitch). The pressure can be controlled by the mod wheel. At high modulation wheel values this knob is used to determine the formant frequency.
	Key-track High	Controls the amount of key-tracking applied to the highpass filter's cutoff frequency. Turn to the right for higher cutoff frequencies at high MIDI pitches.
	Lo Pass 0	Sets the cutoff frequency of an additional lowpass filter within the pipe to enhance formant frequencies. The formants are modified by the pressure in the pipe (not by the pipe's pitch). The pressure can be controlled by the MOD wheel. At low modulation wheel values this knob is used to determine the formant frequency.
	Lo Pass 1	Sets the cutoff frequency of an additional lowpass filter within the pipe to enhance formant frequencies. The formants are modified by the pressure in the pipe (not by the pipe's pitch). The pressure can be controlled by the mod wheel. At low modulation wheel values this knob is used to determine the formant frequency.
	Key-Track Low	Controls the amount of key-tracking applied to the lowpass filter's cutoff frequency. Turn to the right for higher cutoff frequencies at high MIDI pitches.

## Global Controls

The last section of SteamPipe 2 consists of global controls over pitch, polyphony, glide, and an output stage. You also get an Arpeggiator and a test tone generator.

Voice Mode	Pitch Bend	Sets the range of the pitch bend wheel.
	Detune	Introduces a slight detune into the signal for a livelier sound.
	Mode	Menu for the different polyphony modes. Choose between poly, mono, unison, and three arpeggiator modes.
	Glide on / off	Toggles glide on or off.
	Glide Time	This sets the time the pitch of SteamPipe takes to follow incoming MIDI pitches, when [Glide] is on.
	Mod-Whl	This knob follows an incoming mod wheel signal. Use it when you have no hardware controller available.
	Arp Mode	This menu offers different arpeggiation modes. Choose between up (>>), down (<<), up and down (>><<), and random.
	Arp Speed	Menu for choice between different speeds relative to the global tempo.
Output	Spread	Introduces a stereo spread into the main output.
	Gain	Main output volume control.
A440	A440 Tuning tone on / off	If on, a sine oscillator's signal is mixed into the main output. Use it to tune the pipe. The frequency is 440Hz.
	Gain	Controls the volume of the 440Hz tuning tone.

## Space Master Deluxe

You can find this remarkable reverb module on panel B of the SteamPipe 2. Based on several diffusion delays, Space Master can produce a wide array of high-quality natural or experimental ambiances. The patch's useful set of reverb parameters include an early reflections section, a late reflections module, and a post EQ. Dials for main reverb time, control of balance between the two reflection stages, and between dry and wet signal round off the controls.

## Input and output stage

You can put an initial delay into the reverb signal with the predelay [Time] dial and control the predelay's stereo position with the [Symmetry] knob. The [Early / Late Balance] slider can be used to move the source in space – more early reflections bring the signal to the front and more late reflections make it appear further back in space. At the end of the signal chain the [Dry / Wet] slider crossfades between original signal and reverb.

Predelay	Time	Sets an initial delay for the wet signal.
	Symmetry	Introduces a difference into the delay times for the right and left predelay channels. Use this to position the signal around in the stereo field.
Mixing	Early/Late Balance	With this parameter you can set how much of the early and late reflections, respectively, can be heard in the output.
	Dry / Wet	This controls the balance between dry and wet signal.
	Power button	Switches the reverb on or off.

## Reflections

Use the two [Size] and [Diffusion] parameters to control two stages of variable density diffused reflections. The early stage commonly represents the direct response of the virtual space, whereas the late reflections define the sound when the early reflections have died away.

For dynamic reverb effects you can use the Modulation section. It offers an LFO routed to the delay times with [Rate] and [Depth] control. The LFO can enhance your reverb signal by adding liveliness.

Early / Late Reflections	Size	Determines the amount of space generated by the early or late reflections modules by adjusting delay time of the underlying diffusion delays. Higher values give the impression of larger spaces.
	Symmetry	Introduces a stereo shift into the generated reflections.
	Diffusion	Adjusts the perceived density of the reflections generated. Use for a sparser or fuller reverb sound.
	Reverberation Time RT60	This control alters the decay time of the reverb response.

Modulation	Rate	Control of LFO frequency modulating the delay times.
	Depth	Control for the LFO's modulation depth. Higher values yield increased amplitude modulation.

## Frequency response

The two EQ sections serve slightly different needs. The Damping EQs are integrated into the reflection stages and influence their frequency responses. The Post EQ acts on the patch's main output and should be used to finalize the overall sound.

Damping	Low Freq	Cutoff for the low shelving filter that cuts into diffusion delay frequency response of both early and late reflections.
	High Freq	Cutoff for the high shelving filter that cuts into diffusion delay frequency response of both early and late reflections.
Equalizer	Lo Damp	Amount of cut for the low shelving filter.
	Hi Damp	Amount of cut for the high shelving filter.
	Low Freq	Cutoff for the low EQ that acts on the main output of the reverb.
	High Freq	Cutoff for the high EQ that acts on the main output of the reverb.
	Lo Boost	Cut or boost amount for the low Equalizer.
	Hi Boost	Cut or boost amount for the high Equalizer.



# SubHarmonic



SubHarmonic generates pad-like, atmospheric sounds and – at the same time – thick, monophonic lead patches. It consists of two independent sound generators: The [Sub Oscillator] is based on additive synthesis, and the [Formant] section performs like an oscillator with a constant frequency vowel filter. Internally, those oscillators are quite complex; the [Sub Oscillator], for example, does not use normal harmonics of the main frequency to perform additive synthesis, but produces subharmonics. However, those technical details remain hidden below the simple user interface.

## Voice

This section deals with the voice assignment. You have the normal monophonic and polyphonic modes, and an additional unison mode that uses all available voices (like polyphonic) but sets them to the same pitch, only slightly detuned. The resulting phase interferences produce a chorus-like effect.

The amount of portamento and the influence of the MIDI pitchbend wheel onto the instrument can also be adjusted here.

Voice Mode	Controls the voice assignment of the instrument. [Poly] selects polyphonic, [Mono] switches to monophonic behavior; [Uni] is also monophonic, but uses all available voices which are slightly detuned to each other.
Detune	Sets the amount of detuning in [Uni] mode. Turn to the left for larger intervals between the voices.
Glide	Switches portamento on or off (see also [Speed]).
Speed	Adjusts the amount of portamento, i.e. the time that passes until a newly received MIDI note's pitch is gradually reached.
Pitchbend	Sets the range of the MIDI pitchbend wheel in semitones.

## Vibrato

Crucial to this instrument's sound is the vibrato effect. It is produced by mapping an LFO's signal onto the instrument's pitch. The LFO waveform, its frequency, and the amount of pitch modulation can be controlled here, providing settings that differ from the normal idea of musical vibrato but generate impressive sounds nonetheless.

Shape	Selects the waveform of the LFO whose signal is used to modulate the instrument's pitch for vibrato effects.
Rate	Sets the frequency at which the vibrato LFO oscillates.
Width	Sets the pulse width of the vibrato LFO; turn to mid position for a symmetric waveform.
Amount	Sets the amount of vibrato. Turn to the left for no vibrato, to the right for a pitch modulation of the range adjusted by [Range].
Range	Controls the absolute vibrato amount in semitones (see also [Amount]).

## Amplitude and Modulation Envelope

These two envelope generators, placed to the left (modulation envelope) and the right (amplitude envelope) of the [Voice] and [Vibrato] sections, shape the instrument's amplitude and modulate the feedback amount of the [Sub Oscillator] as well as the formant frequency of the [Formant Oscillator]. They operate as normal ADSR envelopes, but offer additional re-triggering options, key-scaling, and adjustable MIDI velocity sensitivity.

Mode	Selects the way the envelope generator reacts to a new gate signal when the previous gate is not yet closed. In [Leg] mode the envelope generator doesn't react to the new gate signal; in [Ret] mode it is re-triggered, using the current envelope level as a starting point; in [Rst] mode the generator is also re-triggered, but reset to its initial level. This control shows no effect if [Voice][Voice Mode] is set to [Poly].
Attack	Sets the attack time of the envelope generator.
Decay	Sets the decay time of the envelope generator.
Sustain	Sets the sustain level of the envelope generator.
Release	Sets the release time of the envelope generator.
Key	Sets the amount and polarity of key scaling applied to the envelope generator's transition times. Turn to the left for faster transitions at low pitches, to the right for slower transitions at low pitches.
Velocity	Sets the amount of MIDI velocity influence on the envelope's amplitude. Turn to the left for constant amplitudes that are independent of velocity, and to the right for full velocity sensitivity.

## Sub Oscillator

This oscillator creates the fundamental MIDI pitch which generates four sub-harmonics below this frequency. The ratio of the harmonics to the main pitch and their amplitudes can be controlled individually, similar to a standard additive synthesizer. A feedback feature provides basic waveform modulation, crossfading smoothly from a sine wave to one that sounds like a sawtooth waveform.

Pitch	Controls the pitch shift of the sub oscillator's main frequency; there is a coarse (top) and a fine (bottom) control. As the sub oscillator generates harmonics below the main frequency, the pitch control shifts those sub tones to a usable frequency.
Harmonic A/B/C/D	Select four sub harmonics of the main frequency. Their volume is controlled by the respective [Amplitude] control.
Amplitude A/B/C/D	Select the amplitude of the corresponding sub harmonic adjusted by the [Harmonic] control.
Feedback Amount	Sets the amount of feedback applied to the sub oscillator internally. Turn to the left for a undistorted sine-like sound, to the right for a saw-like signal.
Envelope Modulation Amount	Controls the modulation amount applied to the [Feedback amount] by the [Modulation Envelope].

## Formant Oscillator

This particular oscillator is made up by a simple sine waveform; however, the adjustable frequency band doesn't move with the oscillator's pitch, but remains stable as a formant of the sound. By moving this frequency band you can achieve vowel filter-like effects.

Pitch	Controls the pitch of the main frequency; there is a coarse (top) and a fine (bottom) control.
Formant Frequency	Adjusts the frequency of the oscillator's formant; this formant is not changed by the main pitch.
Envelope Modulation Amount	Controls the amount and polarity of modulation applied to the [Formant Frequency] by the [Modulation Envelope]. Turn to the left for inverse modulation, i.e. low formant frequencies at high modulation signals, and to the right for normal modulation, i.e. high formant frequencies at high envelope levels.

## Mix and Output

In this section the signals of both oscillators can be mixed, positioned in the stereo field, and leveled.

Mix	Crossfades between the sound of the sub oscillator (at the left) and the signal of the formant oscillator (at the right).
Spread	Sets the amount of displacement within the stereo field applied to the instruments' voices. Turn to the left for a mono signal, and to the right to pan each voice individually; this is particularly impressing if [Voice][Voice Mode] is set to [Uni].
Gain	Sets the output gain.

## Reverb

The high-quality reverb unit is contained within the panel's B view. It can further enhance the sound's spatial characteristics. When not in use it should be turned off by the [Power] control to save CPU power.

Size	Sets the size of the virtual reverberation room.
Symmetry	Places the signal in the virtual reverberation room. Turn to the left or right to move the signal away from the center.
Diffusion	Sets the amount of reverb signal diffusion. Turn to the right for a less echoic sound.
Release	Adjusts the time that passes before the reverberation sound has decayed.
Spin	Sets the amount of modulation applied to the reverb. Technically, the modulation affects the delay time of the delay modules on which the reverb is built.
Frequency	Sets the rate of the LFO used as modulation source (see [Spin]).
High Cutoff	Sets the cutoff frequency of the lowpass filter that is damping the high frequencies.
High Damp	Sets the amount of damping applied to the frequencies above the [High Cutoff] frequency.
Low Cutoff	Sets the cutoff frequency of the highpass filter that is damping the low frequencies.
Low Damp	Sets the amount of damping applied to the frequencies below the [Low Cutoff] frequency.
Mix	Crossfades between the unprocessed, dry signal (at the left) and the reverberated, wet sound (at the right).
Power	Switches the reverb unit on or off. Turn off to save CPU power when the reverb is not in use.

# Grooveboxes

## Aerobic



Aerobic is a step sequencer that controls a virtual analogue drum synthesizer. The instrument produces tight, innovative sounds far beyond the range of traditional drum computers. This, combined with the sequencer's capacities and the mixer's flexible routing options, makes Aerobic a versatile beat production environment that can be used in live performances.

The drum synthesizer contains six similar, independent units (selectable by the tabs at the top of the panel). Each unit combines an oscillator and a noise section into one signal that can be equalized before it is sent to the master mixer. The sequencer (in the middle of the panel) contains two tracks for each sound unit, selectable via the same tabs as the units themselves. The filled rectangles within the sequencer's display represent the unit's trigger signals and their velocity; the unfilled rectangles form a modulation track whose signal can be used to change nearly every parameter of the sound engine and mixer over time: Use the [Modulation] switch in the unit's master section to select the destination of the modulation. The master mixer provides classical mixing parameters for each unit (solo/mute, pan, and, of course, level), along with controllers to adjust the complete ensemble's reaction on MIDI messages. Each unit can be triggered by a selectable MIDI note; on a more complex level, note messages can recall complete ensemble snapshots.

## Sound Engine

The drum synthesizer is built by two sound generators, an equalizer, and a master section that also controls modulation routing. While the oscillator part (on the left side) is based on sine waveforms with frequency modulation capacities, the noise part (on the right side) contains a white noise generator with a multi-mode filter. The mixed signal is sent through an EQ and (within the master section) a final saturator unit before it is passed on to the mixer.

Oscillator	Envelope	Selects the operation mode of the envelope shaping the unit's amplitude. [Lin] activates a standard AD envelope whose transition times are controlled by the [Attack] and [Decay] knobs. When in [Roll] mode, this envelope is re-triggered fast until the next beat; the [Attack] knob in this case also controls the re-triggering frequency. [Roll+Lin] adds both signals of the modes described above. [Noise Env] uses the envelope of the Noise section (see below).
	Attack	Sets the time that passes until the amplitude envelope reaches its peak. In [Roll] mode (see [Envelope]) the knob also controls the rate at which the envelope is re-triggered.
	Decay	Sets the time that passes after the amplitude envelope has reached its peak before it decays to silence.
	Oscillator	Selects the operation mode of the oscillator. While [Sin] represents a standard sine wave, [Sin2] activates a squared sine wave with a different frequency spectrum. Similarly, [FM2] selects the squared signal of [FM] which is generated by a sine oscillator modulating the frequency of another one. (This frequency modulation does not interfere with the modulation controlled by [F-Mod], [F] and [Fmod].) [Phase] uses the output of a phase oscillator.

Mix	F-Mod	Selects the source signal used to modulate the main oscillator's frequency. While [Osc Env] and [Noise Env] select the respective amplitude envelopes, the [Sine], [Tri] and [Random] entries use independent oscillators whose frequency can be adjusted with [Rate].
	F	Sets the base frequency of the main oscillator.
	FMod	Sets the amount of frequency modulation applied to the main frequency by the selected source signal.
	Rate	Sets the frequency of the independent oscillator modulating the main oscillator's frequency.
	Mix	Sets the ratio of the oscillator section's output and the noise section's sound in the signal that is passed on to the equalizer.
	Noise	
	Envelope	Similar to [Oscillator][Envelope], applied to the noise generator's filter.
	Attack	Similar to [Oscillator][Attack], applied to the noise generator section.
	Decay	Similar to [Oscillator][Decay], applied to the noise generator section.
	Noise	Selects the operation mode of the noise section. [White] uses unfiltered noise, [White Mod] modulates the noise generator's algorithm by the noise section's envelope signal.
EQ	Filter	Selects the type of 2-pole filter applied to the noise. Highpass, bandpass, and lowpass filters are available, providing 24 dB damping per octave.
	Freq	Sets the center frequency of the filter.
	Peak	Sets the amount of modulation applied to the filter's center frequency by the envelope.
	Res	Sets the amount of filter resonance.
	Hz	Sets the frequency of the equalizer.
Master	dB	Sets the amount of volume boost (or cut) applied to the adjusted frequency.
	Modulation	Selects the target of the sequencer's modulation track. The modulation shows no effect until the [Track] button is pushed.
	Track	Activates the modulation of the target selected by the sequencer's modulations track.
	Amp	Sets the amplitude of the signal before it is routed to the final shaper unit (see [Shape]).
	Shape	Selects the operation mode of the shaper unit. [Polysat], [Sinesat], and [Hypersat] saturate the signal with tube-like effects; the effect increases the more the signal is amplified before (see [Amp]). [Clean] doesn't perform any compression; [Amp] simply controls the amount of amplification before the signal is routed to the master mixer.

## Sequencer

The sequencer provides two tracks for each of the six drum synthesizer units: a gate pattern and a modulation track. The gate pattern determines the trigger signals and their velocity. The modulation track signal can be routed to any parameter of the sound engine (see [Sound Engine][Master][Modulation]). A roll mode bar provides three different roll modes for fast re-triggering of a drum sound.

Tempo	Selects the tempo of the track: Each step of the sequence can be interpreted as sixteenth note, etc. Thus, the sequencer is always synchronized to the master MIDI clock; use the host sequencer or REAKTOR's internal MIDI clock to start the sequencer. (See also [Global Tempo].)
Global Tempo	Sets the [Tempo] value of all six tracks.
Swing	Sets the amount of swing, i.e. the amount by which every second step of the sequence is delayed to shuffle the strict MIDI rhythm.
Roll Factors	Sets the number of times the trigger signal is repeated if the [Roll Mode] is set to the respective colors.
Init All	Deletes all sequence patterns and modulation tracks and sets [Swing] to its default values.
Track	Selects the track that can be edited within the [Edit Display]
Step Count	Displays the number of the current step (1 to 16). If the gate is off, the number is dark; if the gate is on, the number is light. This can be helpful when editing the modulation track.
Edit Display	Displays the trigger pattern (filled rectangles) as well as the modulation track (unfilled rectangles), depending on the [Track] setting. Clicking within the display allows the patterns to be edited. High values of the trigger pattern represent high velocity; in the modulation track they cause the modulated knob to turn to the right, and low values turn the knob to the left.
Roll Mode	Selects how often the trigger signal is sent. Normally, it is only sent once per beat; by clicking with the mouse one can step through three differently colored modes where the trigger signal is sent more often (see [Roll Factors]).
Loop	Controls the length and position of the played sequence: Only those steps within the rectangle are used. Drag the ends of the rectangle to adjust the loop's start and end points. A second, smaller bar represents the current read out's position.

## Master / Mixer

This section has two functions. First, it mixes the six drum synthesizers down to a single signal – or to four signals if [Single Outs] is activated. Second, it controls the snapshots of the complete ensemble as the sound engine and

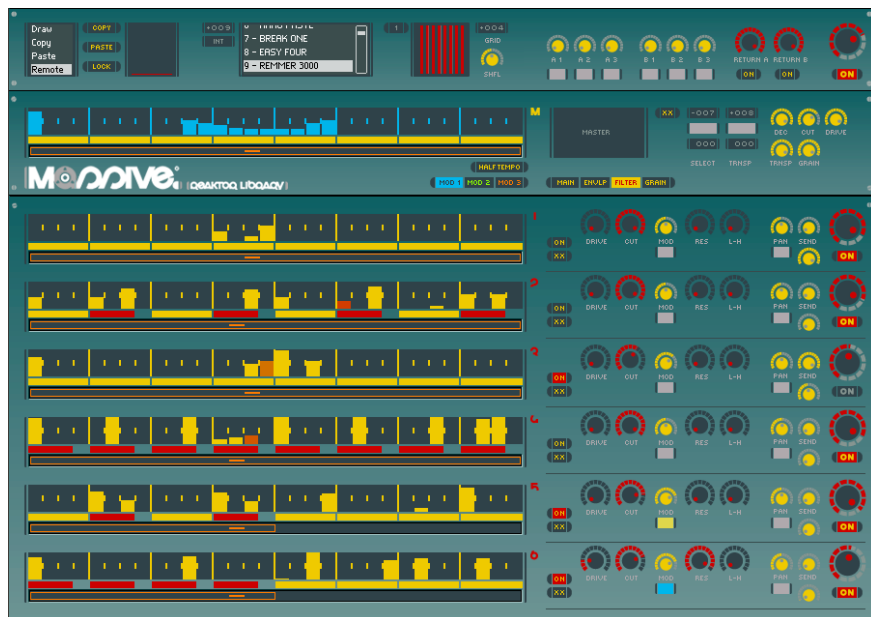


the sequencer are slaved to this part of the instrument. An advanced recall system allows for fast changes of sound/pattern settings via a single MIDI note, making the complex drums computer-controllable from a keyboard in live stage use.

Mixer	Level	Sets the volume of the sound unit.
	Solo	Switches the sound unit to solo playing, i.e. mutes all other units.
	Mute	Mutes the sound unit.
	Pan	Positions the sound unit's mono signal within the stereo field.
	Wide	Enhances the spatial appearance of the sound unit.
Snapshot	Ext. Learn	Activates the learn feature. When pressed, the next MIDI note will be assigned to this track and can be used as external trigger signal, in addition to the internal gate signals of the sequencer. (See also [External].)
	Output	Selects to which of the four stereo outputs the sound unit is routed. This shows no effect until [Single Outs] is activated.
	Power	Switches the snapshot handling on or off.
	Key	Turns on or off snapshot recall by external MIDI note messages. See [Root Note] and [Root Snap] for details.
	Quantize	Turns quantization of external MIDI notes on or off. When on, incoming MIDI messages will be synchronized to a pattern that is selected by [Quantization Select].
	Quantization Select	Selects the quantization pattern to which external MIDI messages can be synchronized.
	Snapshot	Recalls a snapshot of the master mixer. Since all other components are slaved to this one, storing or recalling a snapshot here affects all other instruments, i.e. the sound units and sequencer.
	Root Snap	Sets the snapshot number recalled when the MIDI note adjusted by [Root Note] is received; [Snap Via Key] has to be activated. The note above [Root Note] recalls the snapshot that follows on the [Root Snap] etc.
	Root Snap Learn	The first snapshot recalled after pressing this button will be used as new [Root Snap].
	Root Note	Sets the external MIDI note that, if [Snap Via Key] is on, recalls the snapshot adjusted by [Root Snap].
	Root Note Learn	The first MIDI note received after pressing this button will be used as new [Root Note].
	Store	Stores the current settings of the complete ensemble to the current snapshot number (see [Snap]). If [Store To Next Snap] is activated, the subsequent snapshot number will be used to store the data. Any data previously stored there will be overwritten. Therefore, one should start a completely new bank of snapshots when working on a project.
	Store +1	If activated, upon pressing [Store] the settings of the complete ensemble will not be saved to the current snapshot as displayed by [Snap] but to the next one.

Level	Master	Sets the master volume.
	Velocity	Slaves the [Master] control to the velocity of incoming MIDI notes used to recall snapshots.
	Single Outs	Switches on or off the sound units' routing to different outputs. When off, all sound units are mixed to one stereo signal; when on, there are four stereo outputs to which the six sound units can be individually routed (see [Output]).
	External	Switches on or off the triggering of the sound units by external MIDI notes. When off, the sound units are only triggered by the internal sequencer. (See also [Ext. Learn].)

# Massive



This drum computer is “massive” in at least two ways. First, it contains a vast range of signal-shaping capacities: samples in the six drum tracks don't determine the instrument's sound (like in a standard drum machine), but only provide the material from which the beats can be sculpted. Envelopes, filters, and a potent grain re-synthesis algorithm mangle the fundamental sound until it is completely different, but still musical. Second, these versatile sound editing features are combined with an advanced step sequencer system offering copy and (looped) paste functionality, three different roll modes, a triplet mode, independent loop length for each of the six drum tracks, three modulation tracks whose signal can be routed to nearly every parameter of the sound engine – the list of features could be continued.

Yet, those capacities are not hidden behind an endless array of knobs and faders that prevent productive working. The panel is optimized for usability and fast access to all controllers, making Massive to a powerful sound design workstation. And at the same time – thanks to a complex and glitch-free snapshot recall system – Massive can be used in live performances, or as a slave to a master song sequencer that changes the snapshots automatically.

## Control

At the top of the panel you'll find the instrument's control section. On the left, an edit mode section defines how the various step sequencer displays react to mouse actions. The copy and paste controls are here too. Next to it is the snapshot management system, followed by the quantization and timing controls, and the parameters for external send effects. Finally, there is the master output knob to adjust overall volume.

There are two external effects: a delay unit, and a lo-fi reverb to enhance the sound spatially. Both effects have three parameters that can be controlled from Massive's main panel. Additional parameters can be edited in the effects' own panel, which also contains a normalizer and equalizer. Press {Ctrl}+{2} to go to a second panel set where the effects are displayed; press {Ctrl}+{1} to return to the main panel.

Edit	Edit Mode	Selects the way the various step sequencer displays react to mouse actions. When [Draw] is selected, the mouse can set each step value (see also [Lock] and [Sequencer][Value Display]). When [Copy] is activated, an area of steps can be selected with the mouse that is automatically copied to the [Edit Buffer]. In [Paste] mode the buffer's data is copied back to any area selected with the mouse; if the paste area is longer than the buffer's content, the material to be pasted is looped. [Remote] enables the separate [Copy!] and [Paste!] buttons.
	Copy!	If [Remote] is selected as [Edit Mode], pressing this button activates the same behavior of the step sequencer displays as the separate [Copy] mode of the [Edit Mode]. This button can easily be activated by pressing the {C} key on the computer keyboard (i.e. MIDI note 52). Thus, one can quickly edit the sequencers' data with one hand on the keyboard and one on the mouse. (See also [Paste!] and [Lock].)
	Paste!	If [Remote] is selected as [Edit Mode], pressing this button activates the same behavior of the sequencer displays as the separate [Paste] mode of the [Edit Mode]. This button can easily be activated by pressing the {V} key on the computer keyboard (i.e. MIDI note 53). One can quickly edit the sequencers' data with one hand on the keyboard and the other on the mouse. (See also [Copy!] and [Lock].)
	Lock	Keeps the mouse locked on the selected sequencer step in [Draw] mode (see [Edit Mode]). This can also be activated by pressing the {Z} key on the computer keyboard (i.e.. MIDI note 48).
	Edit Buffer	Displays the content of the buffer into which data is copied in [Copy] mode and that is used in [Paste] mode (see [Edit Mode]).

Snapshot	Snapshot Store	With the left mouse button, a snapshot slot number can be selected; by pressing the right mouse button, the current instrument settings (including all sequencer data) is stored into this slot.
	Snapshot Recall	Displays a list of the available snapshots; selecting a snapshot with the mouse results in recalling its data, including all sequences, but playback is not interrupted.
	Snapshot Mode	Selects whether the snapshots are only recalled via internal signals or if external control signals received at the instrument's [Snap] port are recognized, too. This allows you to connect to a master song sequencer.
Quantization and Timing Effect	Page Switch	Switches between the quantization controls and the timing parameters to be displayed. One page has the [Quantization Select], [Shuffle] and [Grid] parameters, the other contains the [Timing] controls to adjust a micro delay for each track.
	Quantization Select	Selects one of twelve quantization presets. Each preset ranges over sixteen steps; the higher the displayed value, the more delay is applied to this step. The first preset, for example, alternates between low and high values, so every second step will be delayed, resulting in a standard off-beat shuffle. The presets only define relative times; the effective delay time at maximum values is set by the [Shuffle] control.
	Shuffle	Scales the preset of the [Quantization Select] control. Turn to the left for no quantization (independently of the selected preset), to the right for full delay times. See also the [Sequencer] section for interaction with triplet modes.
	Grid	Selects the grid of the sequencer displays; it does not affect the sound.
	Timing	Sets the micro delay for each track.
	A1	Controls the cut-off frequency of the low-pass filter placed within the feedback circuit of the external delay unit.
	A2	Crossfades between pre-delay gating (at the left) and post-delay gating (at the right)
	A3	Gates the signal of the external delay effect. Turn to the left to close the gate, turn to the right to open it. Depending on the [A1] setting, the audio signal will be gated before it is sent into the delay effect or after the effect unit. This parameter can be modulated (see [A/B 2 Modulation Select]).
	B1	Controls the internal cutoff of the spatializer. Use this to alter the color of the effect sound.

Quantization and Timing Effect	B2	Controls the internal resonance of the spatializer. Use this to color the effect sound.
	B3	Gates the signal of the second external effect after the unit. Turn to the left to close the gate, turn to the right to open it.
	A/B 1/2/3 Modulation Select	Selects the modulation track whose signal modulates the value of the respective controller.
	Return A/B	Sets the level of the signal that is returned from the first / second external effect; there is an additional [Mute] button that switches the sound completely off.
Output	Master Volume	Controls the master volume of the instrument. An additional [Mute] button switches the sound completely off.

## Modulation

The three step sequencers of this section don't trigger samples but function as modulation sources to change a sound engine parameter synchronized to the six sample sequencers. Each of the modulation tracks is identified with a color that can be selected within the various modulation source selection controls (e. g. below the Transpose control of the Master section). Normally, you adjust the modulation amount below the source selection control of the modulated parameter.

Track Select	Switches between modulation track 1 (blue), 2 (green) and 3 (orange). All three tracks can be used to modulate various parameters of the instrument, selectable in the various modulation source controls.
Sequencer Display	The modulation tracks' step sequencers act like the one described in the [Sequencer] section; the only difference is the absence of a roll mode.
Half Tempo	Switches between normal and half speed of the resp. track read out: when pressed, each step is interpreted as a eight note; otherwise each step is interpreted as a sixteenth note. (See also [Sequencer][Triplet Display].)

## Sequencer





Each step sequencer display provides three rows of controllers. At the top there is the main display with 32 steps, read out as sixteenth notes. They represent gate values, i.e. the step's velocity. When the gate is completely closed (i.e. the bar is a pulled to the bottom), the sampler unit is not triggered at all.

While the normal gate values are controlled with the left mouse button, holding down the right mouse button and moving the step's value up and down enables rolls, represented as different colors of the step's bar. In roll mode,

the sample is not only triggered once at the beginning of the step, but several times within the step's time.

Below the main step display there is triplet mode control. A yellow bar represents the normal mode, and a shorter red bar appears in triplet mode. Click on the left side of the bar for triplet mode, click on the right for regular mode. In triplet mode, every group of four notes (i.e. steps 1-4, steps 5-8 etc.) can be interpreted as eighth triplets; in this case the last note of the group is not played.

The roll modes depend on the triplet control as seen in the following table:

Duolen	No roll	Sixteenth notes
	Roll 1	Thirty-second notes; the step is triggered twice
	Roll 2	Thirty-second triplets; the step is triggered three times.
	Roll 3	<div> <div> <div>3</div>  </div> <div> <div>3</div>  </div> <div> <div>3</div>  </div> <div> <div>3</div>  </div> </div> <div>Thirty-second note triplets. Every first and second step are quantized differently; setting two subsequent steps to this roll mode results in the displayed pattern. An alternation of this roll mode and no roll mode for two subsequent steps, combined with a regular shuffle of about 66%, thus leads to sixteenth triplets.</div>
Triplet	No roll	Eighth triplets.
	Roll 1	Sixteenth triplets; the step is triggered twice.
	Roll 2	Thirty-second triplets; the step is triggered four times.
	Roll 3	Similar to Roll 2.

In the last row each track has a loop control, dictating which sequencer area loops when the instrument plays. Click and drag the mouse to select the loop area. A small marker indicates the current read-out position within the loop.

Value Display	Displays the main sequence steps. The height of each step's bar represents the velocity of the gate signal produced when the sequencer is running. Depending on the [Control][Edit][Edit Mode], the values can be drawn manually, copied to the edit buffer, or pasted from that buffer using the left mouse button. Movements with the right mouse button pressed activate the roll mode for each step independently, changing the step's color.
---------------	---



Triplet Display	Switches between normal read-out (the steps are interpreted as sixteenth notes) or triplet read-out (the steps are interpreted as eighth triplets) of the group of four steps above each controller. In triplet mode, the fourth note is not played. See the explanations above.
Loop Display	Controls the length and position of the loop to be played. A moving marker shows the position of the read-out when the sequencer is running.

## Sound Engine

The sound engine contains a master section and six independent sampler units. The master section is located to the right of the modulation section; it sets global sample select offset and scales other parameters for all samplers simultaneously. If, for example, [Master][Transpose] is set to 12, all transpose controls of the six drum tracks are scaled from 0 to 12 semitones; if the master control is set to 0, transposition is switched off for all tracks.

The parameters that control the samplers are grouped across four pages. The main page contains the sampler module itself where you load sample files; on the panel it is represented by the sample's waveform. Further controls select the sample from the map and adjust the pitch shift. The envelope section controls the sample's amplitude. The parameters of this page can be used to fine-tune the sample, particularly the influence of the gate velocity on attack and decay times. The filter section contains a low-pass and a high-pass filter whose sound can be smoothly crossfaded. The grain section, finally, controls the grain re-synthesis of the sampler; it is only available in the upper three sampler tracks. Here the speed of the sample traversal can be controlled as well as the grain size, which greatly influences the sound of the re-synthesis algorithm.

Master	Sample Select	Adjusts an offset for all [Sampler][Sample Select] controls of the six independent sampler tracks.
	Sample Select Modulation Source	Selects the modulation track that modulates the [Sampler][Select] parameter. (See also [Sample Select Modulation Amount].)
	Sample Select Modulation Amount	Controls the amount of modulation applied to the [Sampler][Select] parameter by the track selected in [Sample Select Modulation Source].
	Transpose	Adjusts an offset for all [Sampler][Transpose] controls of the six independent sampler tracks. (See also [Transpose Scale].)
	Transpose Modulation Source	Selects the modulation track that modulates the [Transpose] parameter. (See also [Transpose Modulation Amount].)
	Transpose Modulation Amount	Controls the amount of modulation applied to the [Transpose] parameter by the track selected in [Transpose Modulation Source].
	Decay Scale	Scales all decay times adjusted independently for each sampler track with the [Sampler][Decay] control.
	Cutoff Scale	Scales all filters cut-off frequencies adjusted independently for each sampler track with the [Sampler][Cutoff] control.
	Drive Scale	Scales all pre-filter saturation drive amounts adjusted independently for each sampler track with the [Sampler][Drive] control.
	Transpose Scale	Scales all sample transpositions adjusted independently for each sampler track with the [Sampler][Transpose] control. (See also [Transpose].)
	Grain Scale	Scales all grain sizes adjusted independently for each sampler track with the [Sampler][Grain] control. The grain parameter is only available for the upper three sampler tracks.
	Reset	Restores the values of all sampler tracks and the master section to default values.

Page Select		Selects the page displayed on the panel, controlling the [Sound Engine][Sampler] section. The [Main] page contains the following parameters: [Select], [Transpose] and [Reset]. The [Envelope] page contains the following parameters: [Mute by Track], [Velocity], [Decay], [Dynamic Attack] and [Dynamic Decay]. The [Filter] page contains the following parameters : [Drive], [Filter Power], [Cutoff], [Resonance] , [Low-pass / High-pass Crossfade] and [Reset]. The [Grain] page is only available for three of the six sampler tracks; it contains the following parameters: [Speed], [Grain] and [Reset]. The controls for [Pan], [Send Level] and [Track Level] are available in every page. Additional modulation controls are available for some of the controls mentioned above.
Sample	Sample Map Editor	Displays the currently selected sample (see [Sample Select]).
	Sample Select	Selects the track's sample played upon a trigger signal of the step sequencer. (See also [Sound Engine][Master][Sample Select].)
	Sample Select Modulation Source	Selects the source that modulates the [Sample Select] parameter. (See also [Sample Select Modulation Amount].)
	Sample Select Modulation Amount	Adjusts the amount of modulation applied to the [Sample Select] parameter by the source selected in [Sample Select Modulation Source].
	Transpose	Sets the amount of transposition applied to the selected sample. For the upper three sampler tracks this transposition does not affect the sample's playback speed (due to the underlying grain resynthesis algorithm); for the three lower sampler tracks it also changes the playback speed. (See also [Sound Engine][Master][Transpose] and [Sound Engine][Master][Transpose Scale].)
	Transpose Modulation Source	Selects the source that modulates the [Transpose] parameter. (See also [Transpose Modulation Amount].)
	Transpose Modulation Amount	Adjust the amount of modulation applied to the [Transpose] parameter by the source selected in [Transpose Modulation Source].
	Velocity	Adjusts the amount of influence of the sequencer's gate velocity on the sample's amplitude. Turn to the left for no influence, i.e. constant maximum amplitude at every gate value; turn to the right for a complete mapping of the gate value onto the sample's amplitude.
	Decay	Sets the decay time of the amplitude envelope triggered by a gate event. (See also [Dynamic Decay] and [Sound Engine][Master][Decay Scale].)

Dynamic Attack	Sets the amount of modulation by the source selected by [Dynamic Source] applied to the attack time of the amplitude envelope. Turn to the left for no modulation; turn to the right for long attack times at high modulation signals.
Dynamic Decay	Sets the amount of modulation by the source selected by [Dynamic Source] applied to the decay time of the amplitude envelope. Turn to the left for no modulation; turn to the right for long decay times at high modulation signals. (See also [Decay].)
Dynamic Source	Selects the source track that modulates the amplitude envelope's attack time and the [Decay] parameter.
Mute By Track	Selects the mute track. If the specified track receives a gate signal, this track's gate is closed. The feature is particularly useful for programming hi-hats, e.g. the first track plays a closed hi-hat and the second one plays an open hi-hat -- since both tracks mute each other, an open hi-hat sound will be muted when the closed hi-hat sample is triggered. There is also a bypass option to exclude the track from muting.
Drive	Sets the amount of pre-filter saturation drive. (See also [Sound Engine][Master][Drive Scale].)
Filter Power	Switches the track's filter on or off.
Cutoff	Controls the cut-off frequency of the track's filter. (See also (See also [Sound Engine][Master][Cutoff Scale].)
Cutoff Modulation Source	Selects the sequencer track whose signal modulates the [Cutoff] parameter.
Cutoff Modulation Amount	Sets the amount of modulation applied to the [Cutoff] parameter by the source selected by [Cutoff Modulation Source].
Resonance	Sets the resonance of the track's filter.
Lowpass / Highpass Crossfade	Fades between the signal of a low-pass filter (at the left) and the sound of the high-pass filter (at the right). Both use the frequency and resonance adjusted by [Cutoff] and [Resonance].
Speed	Sets the speed of sample read out. This is only available for the top three sampler tracks, which use grain re-synthesis.
Speed Modulation Source	Selects the sequencer track whose signal modulates the [Speed] parameter. This is only available for the top three sampler tracks, which use grain resynthesis.
Speed Modulation Amount	Sets the amount of modulation applied to the [Speed] parameter by the source selected by [Speed Modulation Source]. This is only available for the top three sampler tracks, which use grain resynthesis.

Grain	Sets the grain size of the re-synthesis algorithm. This is only available for the three upper sampler tracks. (See also [Sound Engine][Master][Grain Scale].)
Reset	Sets all controls of the respective controller page to their default values.
Pan	Controls the position of the track's signal within the stereo panorama.
Pan Modulation Source	Selects the sequencer track whose signal modulates the track's [Pan] parameter.
Send Level	Sets the volume of the sampler's signal sent to the external effects. (See also [Control][Effect][Send A/B].)
Track Level	Sets the volume of the sampler's signal sent to the main output. (See also [Control][Output][Master Volume].)

# Newscool



Newscool is a REAKTOR classic– now it's completely rebuilt, with an innovative sequencer (at the top) and the characteristic sound engine (at the bottom). The engine consists of a tone generator on the left and a multi-effect unit on the right. The signal is produced by eight parallel oscillator units whose parameters are modulated extensively. The effect unit parameters – providing pitch shifting, delay and filter - are similarly modulated.

The sequencer is based on the Life model developed by John Conway in the 1970s. A two-dimensional pattern is processed in steps: An element of the pattern becomes alive (dark in this implementation) in the following step if three of its eight neighbors are alive in this step; it remains alive in the subsequent one if two or three neighbors are alive in the current one – else it dies (and becomes a light square again). Several patterns emerge over time by this

set of rules: Gliders move over the grid, crosses oscillate in several phases, some objects remain stable and don't change from step to step while others remain unstable forever. These patterns trigger the sound engine, generating "lively" sequences.

## Life Sequencer

As explained above the sequencer proceeds from one step to the next one by a set of Life rules that translate the current pattern into the following one. The two-dimensional Life pattern is mapped onto the eight channels of the tone generator by the grid of the [Performer Display]: By using the [Wrap X/Y] controllers this mapping can be modified smoothly. The [Sensitivity] knob also interacts with the trigger signals.

Within the [Board Display] Life patterns can be loaded from a bank of factory presets. These patterns can be altered, or you can build completely new ones. The [Board Display]'s content can be copied to the [Performer Display] manually, at the beginning of the Life evolution or at the beginning of each loop.

Loop Display	Shows the process of the loop steps. (See also [Run] and [Length].)
Offset	Sets an offset in steps to the sequencer read-out.
Length	Adjusts the length of the loop in steps. Since the pattern of the [Board Display] can be copied automatically to the [Performer Display] at the beginning of each loop cycle, the loop length controls how often the performer resets to the initial pattern.
Step	Selects the step length of the life sequencer in MIDI units, e.g. selecting a sixteenth calculates a new pattern life phase each sixteenth of the MIDI clock.
Run	Switches the life process on or off. When on, each MIDI clock step (see [Step]) generates a new phase of the pattern according to the life rules (see the instrument description); the result is displayed in the [Performer Display]. The MIDI clock has to be running, or else this button shows no effect.
Next	Calculates the next life sequencer phase independently of the MIDI clock.
Copy	Selects at which point the pattern of the [Board Display] is copied to the [Performer Display]: manually (by pressing the [To Performer] button), at the start of the sequencer when the [Run] button is pressed, or at the beginning of each loop cycle (see [Length]).
To Performer	Copies the pattern of the [Board Display] to the [Performer Display].
To Board	Copies the pattern of the [Performer Display] to the [Board Display].

Board Display	This is a buffer where life patterns can be loaded from the preset list (see [Presets]), edited, or randomly generated. You can draw patterns directly into the display with the mouse.
Presets	Selects a pattern from a list of factory presets, which can then be loaded into the [Board Display] by pushing the [Load] button.
Load	Copies a pattern from the list of factory presets into the [Board Display].
Clear	Deletes the current pattern of the [Board Display].
Random	Randomly generates a pattern within the [Board Display].
Size X/Y	Sets the size of the [Board Display]. When the pattern is copied to the [Performer Display], the size parameters are also adapted to the performer.
Performer Display	Shows the current life phase; its pattern is also used to calculate the next phase. It cannot be edited, patterns can only be copied to it from the [Board Display] (see also [Copy] and [Length]). The grid behind the pattern is used to map the two-dimensional pattern onto a one-dimensional rhythmic sequence (see [Wrap X/Y]).
Wrap X/Y	Controls the projection of the pattern onto the audible sequence; the ratio between horizontal and vertical wrap parameters is visible as a grid within the [Performer Display].
Offset	Adds an offset to the [Wrap X/Y] parameters, thus altering the sequence by shifting it in time.
Sensibility	Determines how many trigger signals are generated from the pattern of the [Performer Board]. Turn to the right for dense trigger sequences, turn to the left for the opposite effect.

## Newscool

The sound engine consists of a tone generator (in the parameter list below referred to as TG) and a multi- effect unit. Both achieve their characteristic sounds via vast modulation of their parameters by two simple LFOs. Those parameters control eight independent synthesizer tracks that are triggered by the [Life Sequencer]; each of the tracks can be muted. The [Random] button sets all those parameters to random values; within the [TG / Effect] Poly Control] areas they can be controlled manually. The parameter shown within these displays is selected using [TG / Effect Parameter Select] controls.

TG Poly Control	Sets the parameters for the tone generator. There are eight bars, one for each track; the value can directly be drawn into the display. The parameter displayed is selectable by [TG Parameter Select].
TG Mute Track	Switches the tracks' tone generators individually on or off.



TG Parameter Select	Selects which parameter of the tone generator is displayed and edited within [TG Poly Control]. There are six parameters available: Pitch, Kick Amount, Frequency Modulation Amount, Ring Modulation Amount, Decay Time and Amplitude.
TG Parameter Modulation	Displays the modulation value for each parameter; by clicking into the display the modulation of the respective parameter can be switched on or off. For modulation, a sine LFO is used (see [TG Modulation Rate/Depth/Phase]).
TG Modulation Rate	Sets the speed of modulation in sequencer steps.
TG Modulation Depth	Sets the amount of modulation.
TG Modulation Phase	Sets the phase of the sine LFO.
Pitch	Sets the absolute range of the pitch modulation. This is a bipolar control: turn the knob to the left for inverse modulation and to the right for normal modulation. There are individual (relative) values for each track adjustable in the [TG Poly Control].
FM	Sets the absolute amount of frequency modulation. There are individual (relative) values for each track adjustable in the [TG Poly Control].
Decay	Sets the absolute decay time. There are individual (relative) values for each track adjustable in the [TG Poly Control].
Drive	Sets the amount of saturation drive applied to the tone generator's signal.
Effect Poly Control	Sets the parameters for the tone generator. There are eight bars, one for each track; the value can directly be drawn into the display. The parameter displayed is selectable by [Effect Parameter Select].
Effect Mute Track	Switches the tracks' effect units individually on or off.
Effect Parameter Select	Selects which parameter of the effect unit is displayed and edited within [Effect Poly Control]. There are six parameters available: pitch shift amount, pitch shift grain size, pitch shift delay time, filter frequency, decay time, and amplitude.
Effect Parameter Modulation	Displays the modulation value for each parameter; by clicking on the display the modulation of the respective parameter can be switched on or off. A sine LFO is used for modulation (see [Effect Modulation Rate/Depth/Phase]).
Effect Modulation Rate	Sets the speed of modulation in sequencer steps.
Effect Modulation Depth	Sets the amount of modulation.

Effect Modulation	Sets the phase of the sine LFO.
Phase	
Filter	Sets an absolute offset to the effect's filter frequency, shifting the individual values of each track that can be edited in the [Effect Poly Display].
Feedback	Sets the level of the signal that is routed from the effect's output back to its input.
Decay	Sets an absolute offset to the effect's decay time, shifting the individual values of each track that can be edited in the [Effect Poly Display].
Mix	Controls the ratio between the unprocessed, dry sound (at the left) and the effect's wet signal (at the right).
Level	Sets the instrument's master level.
Mute	Mutes the complete instrument.
Random	Randomly sets all parameters of each track within [TG Parameter Display] and [Effect Parameter Display].

## Sinebeats 2



The REAKTOR library classic Sinebeats has undergone an overhaul for REAKTOR 5. Sinebeats is a beatbox based on three sine oscillators and a noise generator. Its synthetic nature in combination with the flexible effects section has made Sinebeats a classic for electronic sequence production. Each of the four instruments features a sequencer and individual sound parameters including distortion and filter. Two flexible filters and two delays which are fed via a send/return feature in the mixer add even more motion to the generated beats.

In its new incarnation, Sinebeats got an enhanced mixer with the possibility of routing the individual sound units to single outputs, a two-band equalizer, and a simple compressor for the sum. The sequencers got updated with individual looping, individual clock settings, and the possibility to introduce rolls for every step. You can also record the pitch information via MIDI input. Modulation of sound parameters has undergone a major overhaul, the sine instruments now have multimode filters, and all the instruments now sport an individual overdrive section and an equalizer. A valuable addition to the effects section are the two modulation sequencers that provide dynamic effects sequencing. Also, there is a new snapshot system that enables you to trigger complete snapshots including the sequencer tracks via MIDI note triggers and in sync with the global tempo.

## Sequencer

Each of the four instruments is equipped with its own 16-step sequencer with 2 tracks. The first contains the triggers for the sound units. The second track sends modulation data which can modulate different sound parameters in the instrument. A great addition to your sequencing options is the roll/slide track of the sequencer. You can define 2 different rolls per sound unit that can be assigned to individual steps and you also can introduce pitch slides between steps. The sequencer functionality also allows various direction modes, individual tempo settings and individual loop control. Pitch recording via MIDI input has also been added. If you want to hear Sinebeats without triggering the snapshots via MIDI notes (see [Snapshot system]), you have to switch off [Velocity] in the [Master] section.

Sound units	Switch the view between the [Noise] percussion and the four [Sine] synthesizers and their corresponding sequencers.
Rec	Activates velocity/modulation recording Hit [Rec] and let the sequencer run. Then play notes on your keyboard to write triggers and velocity values for the instruments. For the [Sine] units the note pitch information will be written into the [Pitch dials]. Recording does not delete existing events. They remain untouched as long as no new data is coming in via MIDI.
Run	Starts or stops the sequencer.
Pitch dials	Dial in the desired note pitch per step for the [Sine] units. You can also record the values via midi note input (see [Rec]). These are not available for the [Noise] unit.
Init	Completely initializes the displayed unit's sequencer. This includes deletion of the modulation and velocity tracks, and, in the case of the [Sine] sequencers, resets the Pitch dials.
Direction	Choose between four different direction modes: forwards (->), backwards (<-), and two ping-pong modes (<->, inverted: >-<).
Tempo	Choose an individual tempo for the currently displayed unit sequencer. These are clock division settings that always keep the sequencer in sync with the global tempo.
Loop bar	The bar above the sequencer grid is for setting up a loop region for the currently displayed sequencer. Drag the start or the end to change length and drag the bar to change position.
Random	Randomizes the modulation- or velocity track of the displayed unit sequencer, respectively.

Track selector	Switches the view between the modulation- and the velocity track.
Event grid	Click into the grid and drag up or down to create modulation events or velocity triggers. Right-click (ctrl-click if you're on a Mac) to delete events.
Roll settings	Assign the roll speeds for three freely assignable roll mode colours (yellow, blue and red). You can assign roll overdrive values between 2-times and 16-times. In case of the [Sine] units, the red roll is used for pitch slides only.
Roll and slide modes	The bar below the sequencer grid is for defining roll modes for the individual steps. Left-click to create a roll and repeat the click to change the mode. Right-click (ctrl-click if you're on a Mac) to delete the roll. You can define three roll modes under [Roll settings]. For the [Sine] unit sthe red mode always stands for a pitch slide, with the [Nois] instrument you can freely define it as a third roll mode.

## Noise synthesizer

The noise unit features a simple envelope for controlling the volume. You can stack the different outputs of the multi-mode filter and you can also adjust the parameters cutoff, resonance and envelope modulation intensity. Release, cutoff and resonance can be modulated by the second sequencer track. An overdrive / bit reduction effect and a small equalizer are also at your disposal.

Amp-release	Release time of the amplitude envelope.
Release mod	Bipolar (-/+ ) amount for the modulation track output of the sequencer, altering release time. This can be initialized with the [init mod] button.
Init mod	Initializes all [MOD] controls of the [Noise] unit.
Filter stack	Switches for the output of six different filters. The outputs can be stacked. Since the addition of six signals can produce clipping, the overall level of the instrument will be divided by the number of activated filter signals.
Drive switch	The [Drive] section reduces bit depth and sample rate of the signal, and includes a saturator. Switch on or off with the power button.
Drive	Controls overdrive intensity.
Bit	Controls bit depth reduction.
EQ switch	Switches the 1-band equalizer on or off.
Freq	1-band EQ frequency.
Amt	Bipolar EQ cut / boost. -/+ 24 db.
Cutoff	Filter frequency of the noise filter. Shown in pitch values.

Cutoff mod	Bipolar (-/+) amount for the modulation track output of the sequencer, altering the cutoff frequency. Can be initialized with the [init mod] button.
Reso	Resonance of the noise filter.
Reso mod	Bipolar (-/+) amount for the modulation track output of the sequencer, altering the resonance. Can be initialized with the [init mod] button.

## Sine synthesizers

The three sine instruments are structured identically. Each of them features a release parameter for the decay time, a pitch parameter and a simple pitch envelope with an intensity- and a release parameter. Here you can modulate the intensity of the pitch envelope and the decay time. Pitch can also be slurred with variable glide. As with the noise instrument, you can use an overdrive / bit reduction effect and a small equalizer. You also get a multi-mode filter with variable cutoff and resonance. Both parameters can be targets of the modulation track of the sequencer.

Amp-release	Release time of the amplitude envelope.
Release mod	Bipolar (-/+) amount for the modulation track output of the sequencer, altering release time. Can be initialized with the [init mod] button.
Init mod	Initializes all mode controls of the [Sine] unit.
Glide	Controls glide time. This only works when the sequencer reaches a red [Roll] step. Refer to the description of the sequencer.
Octave	Master octave of the Sine unit.
Tune	Pitch for the sine oscillator of the Sine unit.
Fine / Integer	Toggles pitch control of the sine oscillator between fine and integer mode. In fine mode the range of the dial is +/- 100 cents. These get added to or subtracted from the integer value chosen in integer mode.
Penv	Controls the amount of a percussive pitch envelope applied to the sine oscillator.
Penv mod	Bipolar (-/+) amount for the modulation track output of the sequencer, altering pitch envelope amount. Can be initialized with the [init mod] button.
Prel	Controls the release time of the pitch envelope.
Prel mod	Bipolar (-/+) amount for the modulation track output of the sequencer, altering pitch envelope release time. Can be initialized with the [init mod] button.
Drive switch	The [Drive] section reduces bit depth and sample rate of the signal. Switch on or off with the power button.
Drive	Controls overdrive intensity.

Bit	Controls bit depth of the sound. Reduce to introduce harsh aliasing.
Eq switch	Switches the 1-band equalizer on or off.
Freq	Control of EQ frequency.
Amt	Bipolar EQ cut / boost. -/+ 24 db.
FilterMode	Click multiple times to switch through the available filter modes. You can choose between low-pass high-pass and band-pass.
Cutoff	Filter frequency of the noise filter. Depicted in pitch values.
Cutoff mod	Bipolar (-/+) amount for the modulation track output of the sequencer altering the cutoff frequency. Can be initialized with the [init mod] button.
Reso	Resonance of the multi-mode filter.
Reso mod	Bipolar (-/+) amount for the modulation track output of the sequencer, altering the resonance. Can be initialized with the [init mod] button.

## FX 1 & 2

You get 2 effects units that receive their input from the sends of the [Mixer]. The effects units are identical and offer a stereo delay with an integrated resonant multimode filter, feedback and a return level dial. The filter cutoff can be modulated by an integrated three-waveform, tempo-synced LFO. Both effects units have a small step-sequencer including loop- and tempo control that allow for bipolar modulation of the filter parameters, of [Return level], and [Feedback].

DLY / R	Delay Time for the right channel. Units are beats per echo.
DLY / L	Delay Time for the left channel. Units are beats per echo.
CUT	Cutoff frequency of the multimode filter.
CUT MOD	Switches on modulation of the cutoff frequency by the effects unit's modulation sequencer.
RES	Resonance control of the multimode filter.
RES MOD	Switches on modulation of the resonance by the effects unit's modulation sequencer.
Filter mode	Click multiple times to switch through the available filter modes. You can choose between low-pass high-pass and band-pass. Right-click (ctrl-click if you're on a Mac) to choose the low-pass mode directly.
TEMPO	Dial in the tempo of the LFO. Units are fractions of one bar in quarter notes.
TEMPO MOD	Switches on modulation of the LFO tempo by the effects unit's modulation sequencer.

AMT	Control of LFO modulation depth.
AMT MOD	Switches on modulation of the LFO depth by the effects unit's modulation sequencer.
LFO waveform	Choose between sine, pulse and triangle for the LFO waveform.
Modulation sequencer	Click into the sequencer and drag the mouse up and down to change the value of the bipolar sequencer steps. The sequencer output can be routed to the filter [Cutoff] frequency, filter [Resonance], LFO [Tempo], LFO [Amount], and [Feedback].
RANDOM	Click to randomize the steps of the modulation sequencer.
Loop bar	With the loop bar you can define a region in the sequencer that gets repeated. Drag the start or the end to change length and drag the bar to change position.
DIR	Choose between four different direction modes: forwards (->), backwards (<-), and two ping-pong modes (<->, inverted: >-<).
TEMPO	Via this menu you can choose a tempo for the effects unit's modulation sequencer. These are clock division settings that keep the sequencer always in sync with the global tempo.

## Mixer

The four-channel mixer provides control over [Pan], [Volume], and two effects [sends]. It has a routing system, allowing you to send the different channels either to the master stereo bus or into up to 4 individual stereo busses. This output routing system has to be activated in the [Master] section. The four stereo channels are identical in function and carry the signals of the [Noise] synthesizer and the three [Sine] synthesizers.

Power	Switches the respective channel on or off. Use to mute single or multiple sound units.
PAN	Dial in the position of the respective sound unit in the stereo field.
VOL	Volume of the respective sound unit.
FX 1	Send level to effects unit 1.
FX 2	Send level to effects unit 2.



Output busses	Click multiple times to choose the 4 available stereo busses. Works only if [use single outs] is activated in the [Master] section. Also, this is sensible only if your sound hardware has multiple outputs, or if you route the individual outs of Sinebeats into further REAKTOR instruments. Right-click (ctrl-click if you're on a Mac) to reset to outputs 1 / 2.
------------------	---

## EQ and compressor

With Sinebeats 2 you also get a little effects section that works on the sum. In terms of signal flow it sits between the [Mixer] and the [Master] section. A two-band shelving equalizer and a simple compressor help to spice up the sum of your Sinebeats tracks.

EQ	EQ Power	Switches the sum-EQ on or off.
	F-LOW	Frequency of the low-band shelving EQ. Units are Hz.
	LOW AMT	Cut or boost for the low-band shelving EQ (+/-20 db).
	F-HIGH	Frequency of the high-band shelving EQ. Units are Hz.
	HIGH-AMT	Cut or boost for the low-band shelving EQ (+/-20 db).
Compressor	Compressor power	Switches the sum compressor on or off.
	comp	Dial in the threshold and ratio of compression. These two parameters are combined into one.
	speed	Control for the release time of the compressor.
	soft	If on, the compressor works in soft-knee mode, meaning that the ratio increases gradually to the selected [comp] level. If off, the compression is applied to only to signals above the threshold.

## Master

The Master section provides control over master volume, a switch for velocity sensitivity of triggering the sequences via MIDI input, and a toggle for activation of the multiple output routing system.

Master	Controls the master volume of the patch.
Velocity	Toggles velocity sensitivity for triggering of snapshots via MIDI. If you want to hear Sinebeats' output without MIDI triggering switch this control and the [Snap via key] toggle off.
Use single outs	Activates the three additional stereo outputs. You can route sound into them with the [Mixer].

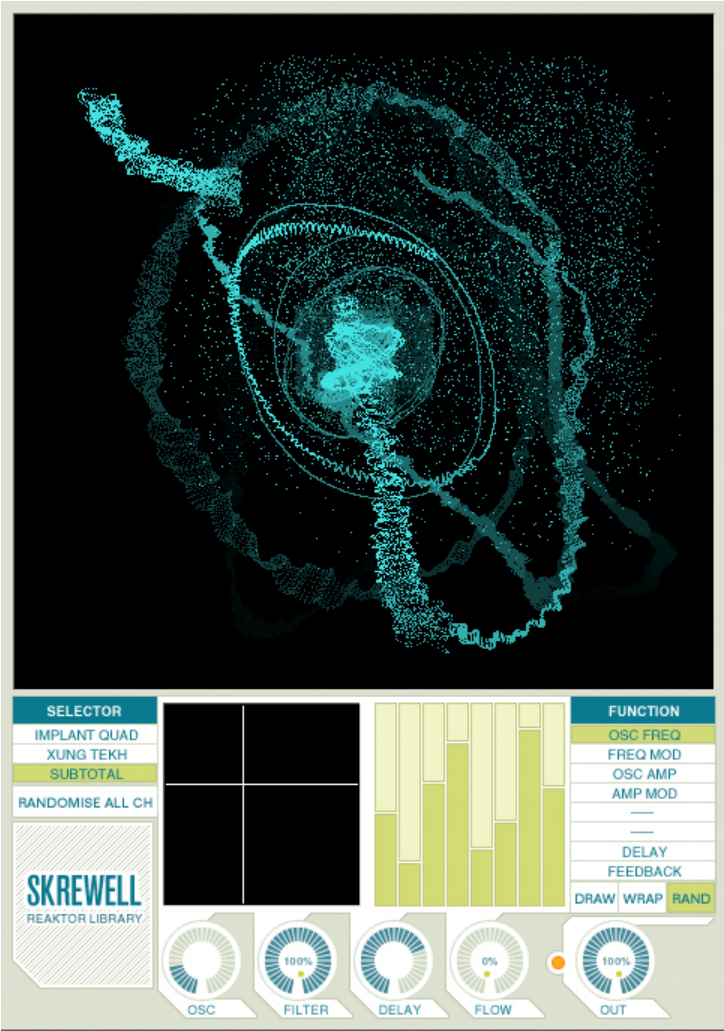
## Snapshot system

The snapshot system is a new feature of Sinebeats 2, enabling you to store and recall snapshots from within the patch. The most intriguing feature of this module is the [snap via key] function. When this is active you can trigger complete stored sequences including all sound units via incoming MIDI note data. This happens glitch-free and in real-time. Use it to trigger Sinebeats sequences in a live situation via a MIDI controller. You can also trigger sequences from another sequencer.

On	This is the bypass switch for the snapshot calling/storing system.
Snap via key	Enables snapshot calling via pitch input.
Start note	Dial in the note that mapping of snapshots across the keyboard starts with.
Learn start note	The first MIDI note value received after pressing this button will be used as new [Start note].
Start snap	Sets the snapshot number that is recalled when the MIDI note adjusted by [Root Note] is received; [Snap Via Key] has to be activated.
Learn start snap	The first snapshot recalled after pressing this button will be used as new [Start snap].
Key-sync	Incoming MIDI notes are quantized at the given resolution relative to the global tempo.
Key sync on / off	Enables / disables key-sync of snapshot calling. This quantizes the start of the next triggered snapshot to a metric value between 1/16th and whole notes.
SnapShot	Choose a snapshot number to store to.
Store	Stores the current settings of the complete ensemble to the current snapshot number (see [Snap]). If [Store+1] is activated, the subsequent snapshot number will be used to store the data. Any data stored there before will be overwritten. Therefore, one should start a complete new bank of snapshots when working on a project.
Store+1	If activated, upon pressing [Store] the settings of the complete ensemble will not be saved to the current snapshot as displayed by [Snap] but to the subsequent one.

# Sound Generators

## Skrewell



Skrewell is an intuitive and visual sound design workstation whose soundscapes can range from meditative atmospheres to crackling harshness. Its sound engine uses eight parallel oscillator sections (channels) that blend into a single,

complex signal. This unique construction means that its interface is unlike that of a classic additive/subtractive synthesizer. The [Draw] edit mode allows for standard value adjustment for each channel's parameters. The parameters are represented by eight vertical bars (one bar for each oscillator section), which control the channels' oscillators, integrated filters and feedback delays. The [Wrap] and [Rand] edit modes provide special ways of altering your chosen parameter across all eight channels simultaneously. The Skrewell unit structure – and therefore the available parameters for each channel – is different in each of the three operation modes. Four main knobs manipulate the sound globally, mainly by mapping the individual channels' parameters. Additionally, a large display visualizes the audio output as Lissajous figure.

## Operation Modes

There are three operation modes, each one based on a unique tone generator system. In Implant Quad mode, each channel consists of a pulse oscillator with subsequent feedback delay; within the delay line a normalizer and a filter alter the signal. Xung Tekh is similar, except the filter is placed before the feedback delay. Subtotal uses a parabolic waveform instead of the pulse waveform and omits the filter completely. The parameters of the tone generators are adjusted in the [Sound Engine] section.

Operation Mode	Selects the main way of operation.
Randomize All Ch.	Sets all parameters of all channels to random values. The [Output Volume] should be lowered to prevent unexpected bursts of noise.

## Sound Engine

This section adjusts the parameters of the tone generators. Depending on the [Operation Mode] setting, a list of the currently available parameters is in the [Parameter Select] display. The selected parameter can then be edited within the [Edit Area], where each bar represents one of the eight parallel oscillator units that form the tone generators.

Function	Switches between the various parameters that control the channels. Depending on the [Operation Mode], there are different sets of available parameters. The values of the selected parameter are displayed for each oscillator section in the [Edit Area].
Edit Mode	Selects in which way the instrument interprets mouse movements within the [Edit Area]. [Draw] allows direct adjustment of each bar. [Wrap] lowers / raises all bars simultaneously, keeping their ratio constant. If a value exceeds the value range it is mirrored. [Rand] performs random changes on all eight bars.

Edit Area	Displays the selected parameter, with one bar representing the parameter's value for each of the eight channels. Mouse movements within this area alter those values, controlled by the [Edit Mode].
Display Control	Scales the Lissajous display.

## Master Controls

These master controls either scale the settings of the [Sound Engine] section (e. g. [Delay Time]) or adjust additional tone generator parameters (e. g. [Flow Amount]). As they affect all eight channels of the tone generators simultaneously they can be used to alter the overall sound.

Oscillator Pitch	Modifies the pitch of all channels. Technically, it controls a mapping function that modulates the values adjusted within the [Edit Area] for each channel. Only very high individual settings will result in high pitches if turned to the left, while less high values will be mapped onto low pitches; move to the right for the opposite effect.
Filter Cutoff	Modifies the filter cutoff frequency of all channels; see [Oscillator Pitch] for technical details.
Delay Time	Modifies the delay time of all channels; see [Oscillator Pitch] for technical details. By turning the knob to the left the delay times can be dramatically shortened, resulting in comb filter-like effects.
Flow Amount	Adjusts various amounts of modulation, depending on the selected [Operation Mode], e.g. frequency modulation amount, amplitude modulation amount etc. Like the [Oscillator Pitch], this knob maps the individual channel's value. Turn to the left for less modulation and more inertia, turn to the right for the opposite effect.
Output Volume	Sets the master output volume. As slight variations of Skrewell's parameters might result in extreme volume changes, this control should be handled carefully. There is an additional [Mute] button at the knob's left.

# SpaceDrone



SpaceDrone generates atmospheric pads which range from light rain or howling wind noises to deep and uncanny space sounds. Technically, the instrument is based on 96 parallel voices spread across the frequency spectrum. Each voice consists of a noise generator; the signal's amplitude is shaped by an envelope, its frequency content gets modified by a bandpass filter, and finally it gets positioned in the stereo field.

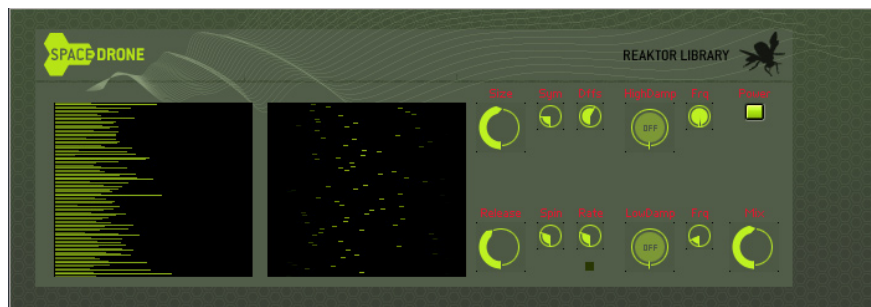
## Sound Engine

The parameters of the sound engine are in the A panel of the instrument. They control the noise generators, their subsequent bandpass filters, the amplitude shaping envelope and corresponding triggering algorithm, and the pan, gain and damping of the signals.

Attack	Sets the time that passes until the amplitude envelope reaches its peak after triggering. The [Density] knob controls speed at which the envelope is re-triggered.
Decay	Sets the time that passes until the amplitude envelope completely fades out after it has reached its peak. The [Density] knob controls speed at which the envelope is re-triggered.
Pitch	Sets the amount by which the amplitude envelope modulates the voice's pitch, i. e. the bandpass filter's center frequency. Turn to the left for inverse modulation – the higher the envelope signal, the lower the pitch. Turn to the right for the opposite effect.
Resonance	Sets the bandpass filter's resonance.
Fundamental	Adjusts the fundamental frequency, i. e. the pitch of the lowest voice.

Offset	Sets the offset of the filter harmonics: All voices are harmonics of the fundamental frequency (see [Fundamental]); all harmonics below the one adjusted here are skipped.
Speed	Controls the rate at which a LFO modulates each voice's frequency randomly.
Amount	Sets the amount by which the voice's frequency is changed by the random LFO.
Density	Sets the speed at which each voice's amplitude envelope is re-triggered.
Random	Sets the randomness of the re-triggering events. Turn to the left for completely regular re-triggering; turn to the right to give each voice a slightly varied re-triggering speed.
Dynamic	Sets the dynamic range of the amplitude envelope. Turn to the left to bind every voice to a constant maximum level; turn to the right to allow some (randomly picked) voices to be quieter.
Pan	Sets the rate at which each voice is rotated within the stereo field.
Random	Sets the randomness of the panning speed. At high values each voice has a slightly different pan rate.
Damp	Sets the amount of damping applied to high frequencies.
Gain	Sets the amount of amplification applied to each voice independently.

## Reverb



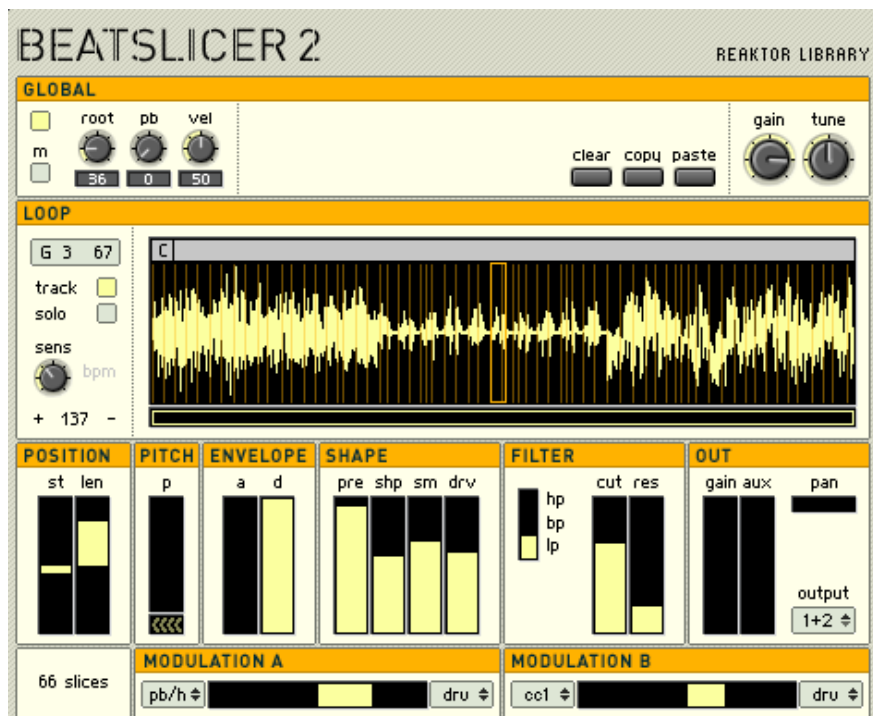
The reverb unit is contained within the panel's B view. It can further enhance the spatial character of the atmospheric pads. When not in use it should be turned off by the [Power] control to save CPU power. Although it is built completely within the new and efficient REAKTOR core layer, it is designed to produce high-quality reverberation sounds.

Size	Sets the size of the virtual reverberation room.
Symmetry	Places the signal in the virtual reverberation room. Turn to the left or right to move the signal away from the center.
Diffusion	Sets the amount of diffusion of the reverb signal. Turn to the right for a less echoic sound.
Release	Adjusts the time that passes before the reverberation sound has decayed.
Spin	Sets the amount of modulation applied to the reverb. Technically, the modulation affects the delay time of the delay modules on which the reverb is build.
Frequency	Sets the rate of the LFO used as modulation source (see [Spin]).
High Cutoff	Sets the cutoff frequency of the lowpass filter that is damping the high frequencies.
High Damp	Sets the amount of damping applied to the frequencies above the [High Cutoff] frequency.
Low Cutoff	Sets the cutoff frequency of the highpass filter that is damping the low frequencies.
Low Damp	Sets the amount of damping applied to the frequencies below the [Low Cutoff] frequency.
Mix	Crossfades between the unprocessed, dry signal (at the left) and the reverberated, wet sound (at the right).
Power	Switches the reverb unit on or off. Turn off to save CPU power if the reverb is not used.



# Sample Player

## BeatSlicer 2



BeatSlicer 2 will separate any waveform into smaller component 'slices', which can then be individually tweaked by adjusting pitch, envelope and FX settings. BeatSlicer 2 is designed primarily for drum-loop manipulation, but the extensive range of parameters offers creative possibilities with any material. For a quick start, right-click (PC) / {Ctrl}-click (Mac) on the large central window, select 'File', 'Load data into table ...' and then choose an audio loop from your sound library. The loop will be scanned and MIDI notes from C-2 (by default) will be assigned to the detected slices.

BeatSlicer 2 is designed to be programmed with MIDI controllers. To assign a MIDI controller to a parameter, use the MIDI learn function on the XY modules on the panel, e.g. the [Pitch] control. You can also assign MIDI controllers to

certain other controls that use Multi Picture modules, such as the [Shape] on/off control. To do this, right click on the module and select 'show in structure'. Then assign a MIDI controller to the hidden button module beneath the Multi Picture module. Be aware that whenever BeatSlicer 2 receives an event from an assigned MIDI controller, it will write to the memory of whatever slice is currently selected. This can be a problem with certain hosts, which send extra MIDI controller data to plug-ins on initialization or when stopping/starting playback. Or it can be a problem if you accidentally move a MIDI controller yourself... The safest thing to do is take regular snapshots of your loop settings.

With the exception of the audio loop itself, all global and per-slice parameters are saved in the host plug-in edit buffer. This means that if you do not change the loaded loop, you do not need to save a new copy of the ensemble. However, as you'll most likely work with different loops in different songs, you should always use the REAKTOR autosave feature. This will create a new copy of the BeatSlicer 2 ensemble and save it with your song.

## Global section

The master controls are located at the top of the instrument panel. They control global settings that are applied to the complete sample loop and not to individual slices. Note that both pitchbend and velocity can be assigned to slices individually using the modulation matrix, in which case it is probably best to set the global knobs to zero.

BeatSlicer 2 has four output channels. By default, these are used as two stereo output channels, but can be used as 4 discrete mono channels by enabling the [Mono] button.

Power	Switches the complete instrument on or off. Technically, this mutes the output; the instrument is still on and consumes CPU power.
Mono	Switches the mono mode on or off. If on, there are four single outs; if off, there are two stereo outputs. (See also [Out][Pan] and [Out][Out Port].)
Root	Selects the root key for the loop – i.e. the first slice will be assigned to this key, (60=Middle C, 48 C-1 etc.).
Pitchbend	Determines the amount by which the pitchbend wheel affects the pitch of the entire loop.
Velocity	Specifies the amount by which note-on velocity affects amplitude for all slices.
Gain	Controls the overall output level in decibels.
Tune	Transpose the pitch of the entire loop in semitones
Clear	Resets all settings of the current slice to their default values.
Copy	Copies all settings of the current slice to an edit buffer (see also [Paste]).
Paste	Copies all settings from the edit buffer into the current slice (see also [Copy]).

## Loop section

The large window displays the waveform of the current loop; it is also the place to load loop files. BeatSlicer 2 can slice loops by employing a transient detection algorithm, or by dividing the loop into equal-length sections. With either method, adjusting the Sens (sensitivity) knob will result in more or less slices. To slice the loop into equal length sections, ensure that the BPM button is active, and that the detected tempo (displayed at the bottom of the loop section) is correct. If the tempo isn't correct, even when adjusted by clicking on the + and - buttons, then the loop will not be an integer number of bars. In this case, you should use the transient detection method (deactivate the BPM button).

Click anywhere on the loop window to select (and audition) selected slices. You can either edit slices individually, or you can choose to edit all slices simultaneously by right clicking on the window. The indicator in the top-left of the Loop section displays the MIDI note assigned to the current slice.

Waveform Display	Shows the waveform of the current loop. To load a different loop, right-click (PC) / ctrl-click (Mac) on the upper part of the window (where the filename is displayed) and select 'File', 'Load data into table ...'.
Zoom Bar	Scrolls the viewable area across the entire loop. Click the right mouse button on this scrollbar to zoom in or out.
MIDI Note	Displays the MIDI note number that is triggering the currently selected slice.
Track	If activated, the received MIDI notes not only trigger the playback of the slices, but also select them for editing.
Solo	If activated, only the currently selected slice can be triggered by MIDI notes.
Sensitivity	Adjusts the amount of slices. If [BPM Switch] is on, this determines whether the loop is sliced into eighth notes, sixteenth notes or thirty-second notes; if the switch is off, the knob controls the sensitivity of the transient detection algorithm.
BPM Switch	Toggles between automated transient detection (off) and a slicing of the loop into parts of equal length (on).
BPM Control	Adjusts the tempo of the loop. A tempo is extracted from the length of the sample file; by using the [+] and [-] controls this value can be modified.

## Slice Parameters

This section below the waveform display adjusts the parameters of the currently selected slice. These parameters control the slice's start position and length, its transposition, an amplitude envelope, a hybrid compression / distortion unit entitled [Shape], and a filter. By clicking their headlines, the [Envelope], [Shape] and [Filter] part of the section can be switched on or off. Note that the envelope section, if switched off for each slice, can still be used as a modulation source.

Position	Start	Adjusts the start position of the current slice.
	Length	Adjusts the length of the current slice.
Pitch	Transpose	Determines the amount of transposition and its direction – up or down.
	Reverse	Switches between forward and reverse playback of the current slice.
Envelope	Attack	Specifies the time taken to reach full amplitude, as a proportion of slice length. Thus, if set to 50%, the envelope will reach peak value halfway through the slice.
	Decay	Determines decay time and shape as a proportion of the time remaining after the attack phase. At maximum, the envelope will sustain at full amplitude for the entire slice (or the remainder of the slice after the attack phase). When between 50% and maximum, the envelope consists of a sustain phase followed by a decay phase. At less than 50%, there is no sustain period, just a decay stage.
Shape	Pre	Increases the compressor input level.
	Shape	Determines the compressor gain curve.
	Smooth	Reduces the amount of distortion by smoothing gain changes as it controls the attack and release of the compressor.
Filter	Drive	Saturates the output signal.
	Mode	Selects the operation mode of the filter unit. Low-pass, band-pass and high-pass modes are available.
	Cut-off	Sets the center frequency of the filter.
Out	Resonance	Sets the resonance of the filter at the cut-off frequency.
	Gain	Adjusts the output level in decibels
	Aux Send	Sets the level of the auxiliary output port of the instrument.
	Pan	Controls the position of the sound within the stereo field.
	Out Port	Selects the output of the instrument to which the slice's sound will be routed. Depending on the global [Mono] switch either two stereo ports or four mono ports are available.

## Modulation

BeatSlicer 2's advanced modulation routing allows various parameters to be modulated by a variety of sources (both MIDI and internal). In both modulation sections (A and B) the left-hand box displays the current source, and the right-hand box displays the current destination. Click and drag vertically to change the source or destination. The slider bar in between specifies the amount (and direction) by which the source modulates the destination. For example, to assign velocity to amplitude, select 'Vel' as the source, 'Amp' as the destination, and set the slider bar to the full right-hand position.

Some modulation sources have a variation denoted by 'H'. This option samples the value of the source when triggered. Try assigning the Pitchbend wheel to Pan and repeatedly trigger the sample while modulating the Pitchbend wheel. Change the source to 'PB/H' and listen for the difference.

Sources	Vel	MIDI note on velocity.	Unipolar
	PB	MIDI pitchbend wheel.	Bipolar
	PB/H	MIDI pitchbend wheel, sampled at note-on.	Bipolar
	CC1	MIDI controller 1 (the modulation wheel).	Unipolar
	CC1/H	MIDI controller 1, sampled at note-on.	Unipolar
	CC7	MIDI controller 7 (the volume slider).	Unipolar
	CC7/H	MIDI controller 7, sampled at note-on.	Unipolar
	Env	Envelope generator.	Unipolar
	Rnd	Random value generator.	Bipolar
Destinations	Amp	Slice amplitude	(-100% to +100%)
	Pan	Stereo pan	(-100% to +100%)
	P	Slice pitch	(-12 to +12 semitones)
	Len	Slice length	(-100% to +100%)
	Drv	Overdrive amount	(-60 to +60 decibels)
	Cut	Filter cutoff	(-120 to +120 semitones)
	Aux	Aux send level	(-100% to 100%)

## Memory Drum 2



Memory Drum 2 is an advanced sampler that enables the independent configuration of up to 128 samples in a compact, easy-to-use interface. Specifically designed for drum sampling, it features an attack-hold-decay envelope, a range of effects, multiple output channels, and complex modulation options. The intuitive interface allows drum kits to be constructed quickly and easily, yet the extensive range of sound-design options offer vast creative possibilities for generating new sounds from your existing samples.

For a quick start, double-click on the sampler window, open the REAKTOR sample map editor, and load Memory Drum 2 with a few drum samples. As you trigger the samples from your keyboard, notice that the green box in the *Edit* section moves to indicate the current MIDI note. Any parameter that you adjust will be stored for that MIDI key. For example, press a MIDI note and then adjust the Envelope attack and decay time. Now press another key, and adjust some parameters for that sample, and so on ...

Memory Drum 2 is designed to be programmed with MIDI controllers. To assign a MIDI controller to a parameter, use the MIDI learn function on the XY modules on the panel. You can also assign MIDI controllers to some parameters that use Multi Picture modules, such as the 'Shape' on/off button. To do this, right click on the module and select 'show in structure'. Then assign a MIDI controller to the hidden button module beneath the Multi Picture module. (You can also assign controllers to the Bank and Sample controls in the Sample section in this way, allowing you to browse through the sample map using MIDI controllers.)

Be aware that whenever Memory Drum 2 receives an event from an assigned MIDI controller, it will write to the memory of whatever sample (or samples) are currently selected. This can be a problem with certain hosts, which send extra MIDI controller data to plug-ins on initialization or when stopping/starting playback. Or it can be a problem if you accidentally move a MIDI controller... The safest thing to do is take frequent snapshots of your drum kit configuration.

With the exception of the sampler map configuration, all parameters are saved in the host plug-in edit buffer. This means that if you do not change the sample map, you do not need to save a new copy of the ensemble. But if you make *any* changes to the sample map, you should use the REAKTOR autosave feature. This will create a new copy of the Memory Drum 2 ensemble and save it with your song. When in doubt, use the autosave feature to avoid data loss.



## Global Parameters

The master controls are located at the top of the instrument panel. They adjust the instrument's global settings, which affect all loaded samples. Note that both pitchbend and velocity can be assigned to samples individually using the modulation matrix, in which case it is probably best to set the global knobs to zero.

Power Switch	Mutes the entire instrument. This does not switch the instrument off to save CPU power.
Mono	When activated, the instrument provides four independent mono channels as output ports; otherwise, it offers two stereo ports.
Bank Number	Sets the number of sample banks. See the [Edit and Sample] section for details.
Shift	Transposes MIDI note input up or down as required.
Pitchbend	Determines the amount that the pitchbend wheel affects the pitch of the entire kit.
Velocity	Specifies the extent to which note-on velocity affects amplitude for all samples.
Tune	Transpose the pitch of the entire drum kit in semitones.
Gain	Controls the overall output level in decibels.
Clear	Resets all parameters of the current note to their default values.
Copy	Copies all parameters of the current note to an internal buffer.
Paste	Copies all parameters of the internal buffer to the current note.

## Sample & Edit

The [Edit] section displays the sample map: Each slot represents a MIDI note; if this MIDI note is received, the sample selected within the [Sample] section is triggered. As there is a maximum of 128 different MIDI notes, normally only 128 samples can be loaded into a REAKTOR sample map.

However, the two selection controls above the waveform display of the [Sample] section – entitled [Bank Select] and [Sample Select] – override this limitation. The best way to explain this is by example. Imagine you have a total of 512 drum sounds on your hard disk (this is just a hypothetical example!), and you wanted to load them all into Memory Drum so that they are all readily available for selection. Start by setting the number of banks to four (using the bank knob in the global section). Next, load the first 128 samples (using the sample map editor), assigning them to MIDI notes 0 to 127, and from velocity 1 to velocity 31. Then load the next 128 samples to MIDI notes 0 to 127, from velocity 32 to 63. Repeat this process for the remaining two 'banks' of 128 samples. You can now select any sample in the map by using the [Bank

Select] and [Sample Select] lists on the panel. Although initial map creation is time-consuming, it can be extremely useful once set-up. Imagine having 128 kick drums loaded into the first bank, 128 snares loaded into the second bank, 128 hi hats into the third bank and so on... This can enable quick and easy kit creation, and convenient auditioning of samples on the fly.

Edit	Sample Map Display	Selects the current MIDI note slot for editing. You can select a range of notes to edit simultaneously by clicking the right mouse button and dragging the mouse. Double-clicking on this bar automatically selects all notes for simultaneous editing (double-click again to return to the previous selection)
	Zoom Bar	Scrolls the viewable area across the entire MIDI note range. Clicking the right mouse button on this scrollbar cycles between 3 different zoom states.
	MIDI Note	Displays the current MIDI note selected for editing within the [Sample Map Display].
	Track	If activated, the received MIDI notes not only trigger the playback of the sample, but also select them for editing.
	Solo	If activated, only the currently selected sample can be triggered by MIDI notes.
Sample	Bank Select	Selects the bank from which the [Sample Select] controller picks a sample.
	Sample Select	Selects the sample that is played when the currently active MIDI note is received (see also [MIDI Note] and [Sample Map Display]).
	Sampler	Displays the wave file selected by [Sample Select]. You can also load new files into REAKTOR's internal sample map editor here.
	Start Position	Adjusts the start position within the sample file.
	Reverse	Switches between forward and reverse playback of the sample.
	Pitch	Transposes the sample up or down in semitones.

# Sample Parameters

In this section you can adjust the parameters and effect settings of the currently selected sample (see [Edit][Sample map Display]). There is an envelope controlling the sample's amplitude, a lo-fi distortion effect, a compression / saturation unit labeled [Shape], a multi-mode filter and a final output part. [Lofi], [Shape] and [Filter] can be toggled on and off for the selected sample by clicking the respective section's title.

Envelope A	Sustain/Release Mode	If activated, the envelope remains at full amplitude after the attack time until the MIDI gate signal is closed; then the decay time is interpreted as release time.
	Attack	Specifies the time taken to reach full amplitude.
	Hold	Specifies the time held at full amplitude.
	Decay	Specifies the time taken for amplitude to fall back to zero.
Lofi	Hertz	Adjusts the re-sampling frequency in Hertz.
	Bit	Adjusts the bit depth of the re-sampling algorithm.
	Mix	Crossfades between the unprocessed, dry signal and the processed, wet sound.
Shape	Noise	Crossfades between the re-sampled signal and a noise generator to be mixed with the unprocessed sound.
	Pre	Increases the compressor input level.
	Shape	Determines the compressor gain curve.
	Smooth	Reduces the amount of distortion by smoothing gain changes as it controls the attack and release of the compressor.
Filter	Drive	Saturates the output signal.
	Mode	Selects the operation mode of the filter unit. You can choose between low-pass, band-pass and high-pass filter modes.
	Cut-off	Sets the center frequency of the filter.
Out	Resonance	Sets the resonance of the filter at the cut-off frequency.
	Gain	Adjusts the output level in decibels.
	Aux Send	Sets the level of the auxiliary output port of the instrument.
	Pan	Positions the sound within the stereo field.
	Out Port	Selects the output of the instrument to which the slice's sound will be routed. Depending on the global [Mono] switch either two stereo port or four mono ports are available.

**Voice Group** By default, REAKTOR rotates voices to minimize voice-stealing. However in the context of drums, voice-stealing can often be desirable. Consider the example of a pair of open and closed high-hat samples – you may want these two samples to share the same voice, so that the triggering the open-hat sample truncates the closed-hat sample and vice versa. Please note that voice groups will only work effectively if: (1) the highest group number in use does not exceed the number of voices in the instrument properties, (which is four by default); and (2) all samples are manually assigned to a voice group (rather than a mixture of auto and manual voice assignments).

**Modulation**

Memory Drum 2’s advanced modulation routing allows various parameters to be modulated by a variety of sources. Beside the MIDI sources – like the modulation wheel and the pitchbend control – there are two envelopes and an LFO. ([Envelope A] is hard-wired to the amplitude of the sample playback but can also be used as freely assignable modulation source.) Some modulation sources have a variation denoted by ‘(hold)’. This option samples the value of the source when triggered. Try assigning the Pitchbend wheel to Pan and repeatedly trigger the sample while moving the Pitchbend wheel. Change the source to ‘Pitchbend/H’ and listen to the difference.

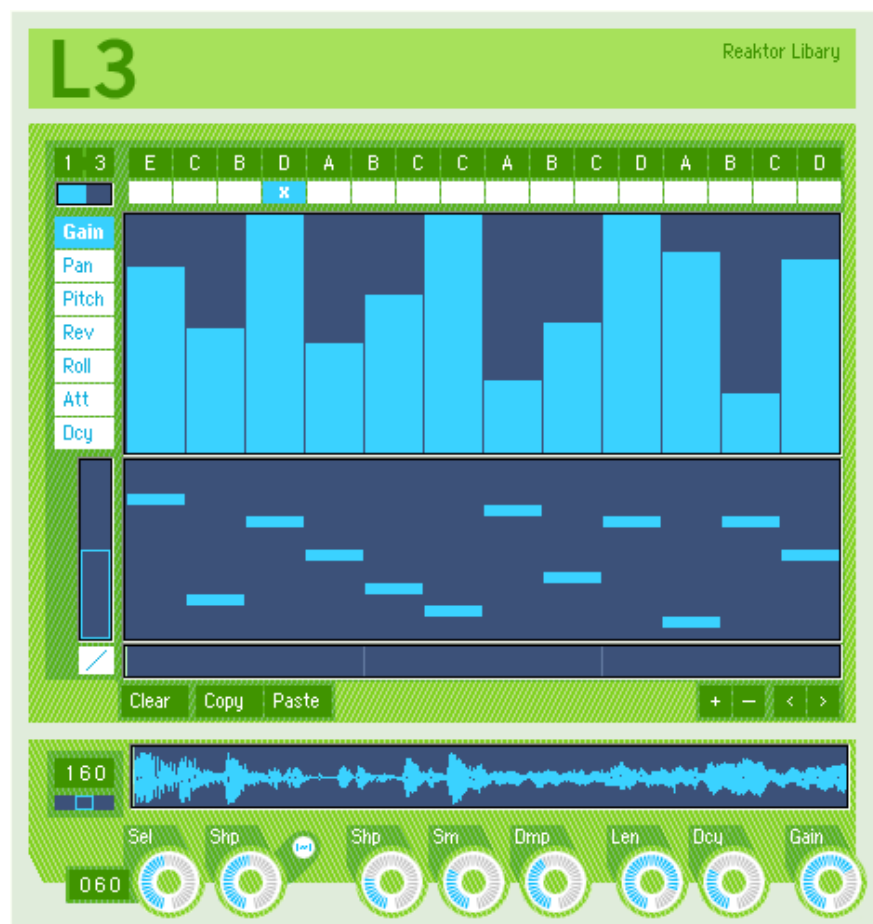
Envelope B	Shape	Morphs the envelope shape from concave (at low values) to linear (at center) to convex (at high values).
	Attack	Specifies the time taken to reach full amplitude.
	Decay	Specifies the time taken for amplitude to fall back to zero.
LFO	Waveform	Selects the waveform of the low frequency oscillator.
	Operation	In [Hz] and [Sync] modes, the LFO phase is reset every time the note is triggered, the only difference being that the frequency is quantized to tempo in [Sync] mode. In [Lock] mode, the LFO frequency snaps to the MIDI tempo and the LFO phase gets locked to the MIDI song position.
	Mode	
	Speed	Sets the rate at which the LFO oscillates.
	Phase	Sets the phase to which the LFO is reset when triggered by note events.

In each modulation section (A, B and C) the top box displays the current source, and the bottom box displays the current destination. Click and drag vertically to change the source or destination. The slider bar specifies the amount (and direction) to which the source modulates the destination. For example, to assign velocity to amplitude, select 'Velocity' as the source, 'Amp' as the destination, and set the slider bar to the full right hand position.

Sources	Velocity	MIDI note on velocity.	Unipolar
	Pitchbend	MIDI pitchbend wheel.	Bipolar
	Pitchbend (hold)	MIDI pitchbend wheel, sampled at note-on.	Bipolar
	CC1	MIDI controller 1 (the modulation wheel).	Unipolar
	CC1 (hold)	MIDI controller 1, sampled at note-on.	Unipolar
	CC7	MIDI controller 7 (the volume slider).	Unipolar
	CC7 (hold)	MIDI controller 7, sampled at note-on.	Unipolar
	Env A	Envelope generator A.	Unipolar
	Env B	Envelope generator B.	Unipolar
	LFO	The LFO.	Bipolar
Destinations			(except Sqr+ & Ramp)
	LFO (hold)	The LFO, sampled at note-on.	Bipolar
	Rnd	Random value generator.	Bipolar
	Amp	Slice amplitude	(-100% to +100%)
	Pan	Stereo pan	(-100% to +100%)
	Pitch	Slice pitch	(-12 to +12 semitones)
	Start	Sample start position	(-1 to +1)
	Env A Attack	Envelope A attack time	(approximately -10 to +10 seconds)
	Env A Decay	Envelope A decay and hold time	(-100% to +100%)
	Hz	Lofi resampling frequency	(-100 to +100 semitones)
	Drive	Overdrive amount	(-60 to +60 decibels)
	Cut	Filter cutoff	(-120 to +120 semitones)
	Aux	Aux send level	(-100% to 100%)
	Env B Amp	Envelope B amplitude	(-100% to 100%)
	LFO Amp	LFO amplitude	(-100% to 100%)

# Sample Transformer

## L3



L3 is a sequenced drum loop recycler: Load a loop, click some stuff, mangle the loop.

The instrument's panel can be broken down into three sections. The top section contains a pattern sequencer and global controls for pattern length and tempo swing. The middle section has the main step sequencer where patterns can be edited (up to eight patterns can be programmed and arranged per

snapshot). The lower section contains the sample playback engine controls (including the sampler window where loops are loaded).

L3 is driven by the MIDI clock. This means that when used in the stand-alone version of REAKTOR, the play button on the REAKTOR toolbar must be pressed. When used as a plug-in, L3 will only run when the host sequencer song is playing.

## Pattern sequencer

Each L3 snapshot consists of up to eight individual patterns labeled A through H. As you will see on the panel, there are 16 [Pattern Selector Boxes] along the top each displaying a letter (A to H). The pattern playback order depends on the arrangement of the letters displayed in these 16 boxes. The leftmost box selects the pattern to be played for the first bar, the second box selects the pattern for the second bar, and so on.

Pattern Selector Boxes	Controls the sequence of the patterns to be played one after another. Click and drag vertically on any of the 16 boxes to select a pattern. Use the [Loop Area Bar] to determine the length and position of the loop.
Loop Area Bar	Defines the area of the [Pattern Selector Boxes] played in a loop. If only one box is selected, only this pattern will be played; this can be useful to edit and audition a pattern.
Bars per Pattern	Adjusts the number of bars in each pattern.
Beats per Bar	Controls the number of beats in each bar. A beat is interpreted as a quarter note; each step represents a sixteenth note.
Swing	Sets the amount of shuffle, i.e. the amount of slight delay on off-beats.

## Step sequencer

Things get more interesting here. L3 features eight parameters which can be sequenced by programming their value at each 16th. This is what turns old loops into new loops...

The most obvious of these eight parameters is slice order. This is displayed in the lower of the two large windows (the [Slice Position Sequencer]). By clicking the mouse here you can rearrange slices of the original loop. The window is 16 steps high which means you can select the first 16 slices of your loop. If the selected loop has more than 16 steps, use the [Scroll Bar] at the left to see more slices. The right-mouse button has a function here too: it restores any step to its default value (i.e. to the original slice order).

The upper window (the [Parameter Sequencer]) is for editing the remaining seven parameters: gain, pan, pitch, reverse, roll, attack, and decay. Clicking the right-mouse button resets steps to their default value.

Parameter Sequencer	Controls the values of the various parameters for each sequencer step. Use the right mouse button to reset a parameter to its default value.
Gain	Adjusts the gain of each slice.
Pan	Adjusts the position of each slice within the stereo field.
Pitch	Adjusts the pitch of each slice, i.e. its transposition in respect to the original pitch of the sample file.
Reverse	Determines the playback direction. At minimum (the default setting), slices will play forward as normal. At any other value they will play in reverse. At lower values, playback will start from near the end of the slice, whereas with higher values, playback will start from nearer the beginning of the slice.
Roll	Causes the slice to retrigger repeatedly within each step. With higher values the slice will retrigger more quickly.
Attack	Causes the loop volume to suddenly cut out and then fade back in. At maximum, the fade in time is exactly 1 beat (i.e. four steps).
Decay	Modulates the envelope decay time. At center (default) decay time is unaffected. With higher values the decay time is extended, and with lower values the decay time is reduced. (Therefore the effect depends on the envelope decay time setting controlled by the [Decay] control of the [Sampler] section.)
Slice Position Sequencer	Controls the order of the slices. Low values represent slices at the beginning of the sample file, high values slices at its end. Thus, a line from the bottom-left to the top-right results in normal playback order as defined by the sample file without any re-arrangement.
Scroll Bar	Scrolls the [Slice Position Sequencer] vertically. This can be useful if a long loop with many slices is loaded: As the [Slice Position Sequencer] can only display sixteen vertical values, slices after the sixteenth cannot be controlled. Use this bar to scroll to those higher values.
Edit Range Bar	Controls the area of steps within the [Slice Position Sequencer] onto which the edit functions are applied. The edit functions are:
Reset Slices	Sets each step within the edit range to its default position.
Shift Up / Down	Shifts each step within the edit range up or down by one position.
Shift Left / Right	Rotates each step within the edit range to the left or right by one position.
Clear	Resets all steps within the edit range in both the [Slice Position Sequencer] and the [Parameter Sequencer].
Copy	Copies all steps within the edit range of both the [Slice Position Sequencer] and the [Parameter Sequencer] into an internal buffer.
Paste	Copies all steps from the internal buffer into the edit range of both the [Slice Position Sequencer] and the [Parameter Sequencer].



## Sampler

The main window is for loading loops and displays the currently selected waveform. After loading a loop, make sure it is selected using the [Sample Select] knob, and then check that the detected tempo is correct (it's displayed in the box to the left of the sample window). If incorrect, the tempo can be adjusted using the slider bar beneath. If the correct tempo cannot be selected, then the loop is not an integer number of bars in length, in which case you cannot use it.

All of the sampler controls in this section are stored per-pattern. Clicking a knob with the left mouse-button writes to the current pattern only, whereas clicking with the right-button writes to all eight patterns simultaneously (A to H). Also, double-clicking on a knob resets it to its default position.

Sample Display	Displays the sample currently selected by [Sample Select]. Double-click to open REAKTOR's Sample Map Editor and to load a sample file.
Tempo Control	Displays the automatically extracted tempo of the sample loop in beats per minute. Use the slider to select a different value.
Sample Select	Selects a sample from the map within REAKTOR's Sample Map Editor of the [Sample Display].
Pitch	Transposes the overall pitch of the loop in semitones.
Stretch	Calculates the pitch at which one bar of the audio file will be the same length as one bar of the actual current song tempo, and then transposes the loop accordingly. In other words when 'stretched', there will be no gaps between slices (caused by the original loop tempo being slower than the current tempo), nor will slices be prematurely truncated (caused by the original loop tempo being faster than the current tempo). It is still possible to transpose the loop when the stretch button is active, but obviously the loop will no longer be perfectly stretched to tempo. In other words, to be correctly stretched the pitch knob must be set to zero.
Shape	Determines the compressor gain curve (see also [Smooth] and [Damp].)
Smooth	Reduces the amount of distortion by smoothing gain changes; it controls the attack and release of the compressor. (See also [Shape] and [Damp].)
Damp	Attenuates high frequencies, reducing 'grainy' sounding compression artifacts. (See also [Shape] and [Damp].)
Length	Sets the hold period (i.e. the length) of the envelope shaping each slice's amplitude.
Decay	Sets the decay time of the envelope shaping each slice's amplitude. This is the master control that can be varied for each step independently.
Gain	Sets the output level for the current pattern.

## Random Step Shifter



Randomstepshifter uses intelligent pseudo-random principles to cut-up and rearrange sample loops, on-beat, in real-time. There's an intuitive three-part sequencer that triggers sample playback. It also modulates sample selection, positional offset, and playback pitch. In addition, these modulations can be mangled by various pseudo-random sequences. This instrument will create new sample loops for you very easily! You can load in any loop, just keep in mind that you need to cut the loops accurately so that they play correctly when they are looped over their entire length.

### SQ2

The Randomstepshifter contains a simple step-sequencer. It consists of three parts, the [Select], [Offset], and [Pitch] sequencers. Each of the parts has two tracks - the trigger track at the bottom of the sequencer, and the modulation track above. The trigger tracks can be used independently from the modulation tracks but you can't modulate without a trigger. In other words, you can trigger the envelope without sending any modulations, but not vice-versa. In the [Envelope] section you can choose which of the three trigger tracks is used for starting the envelope. The trigger events of the [Offset] track can also be used to reset sample offset. The modulation tracks can be used to modulate

the sample player's main parameters. These are the [Select] parameter for sample selection, the [Offset] parameter for controlling the start position in the currently selected sample, and the [Pitch] parameter controlling pitch of sample playback.

Loop bar	The bar above the sequencer grid represents the sequencer's loop region. Right-click (ctrl-click for Mac users) to set length. Left-click and drag to move.
Modulation tracks	Click on the grid to create modulation events. Drag the mouse up or down to set the level. These events are wired to the [Sample select], [Sample offset], and [Pitch] modules, respectively. They can be used to modulate the parameters of the respective modules in a controlled or randomly varied way. These parameters are sample selection, sample offset, and pitch of playback. Right-click (ctrl-click if you're on a Mac) to delete the event, along with its trigger event. You will automatically create [Trigger] steps associated with the modulation events. See [Trigger tracks] for more information. Drag the modulation bar completely down to have a trigger event without modulation output. The three tracks can be selected with three buttons below the sequencer ([Select], [Offset], and [Pitch]).
Trigger track	Click to create events that trigger the [Envelope]. Drag down the corresponding modulation event to zero if you want a sole trigger without modulation. All three trigger tracks can be used to start the envelope. Use the respective buttons in the [Envelope] section to choose which track will do it. Furthermore, the [Offset] trigger track can reset the sample offset if the [Seq] button in the [Sample offset] section is active.
Select / Offset / Pitch	These buttons switch the view to show the three tracks of the sequencer. The modulation part of the [Select] track is wired to the [Sample Select] module, the modulation part of the [Offset] track is wired to the [Sample Offset] module and the modulation parts of the [Pitch] track is wired to the [Pitch] module.
Copy	Copies the current loop region into the clipboard.
Paste	Pastes the pattern clipboard into the current pattern.
Rand	Randomizes the current loop region.
Clear	Clears the current loop region.
Zoom Level (16 st, 32 st, 64 st)	Click and drag mouse up or down to zoom in and out of the currently displayed pattern.
Clock divider (1/6, 1/8, 1/12, 1/16, 1/24, 1/32)	Choose between different clock divisions. This speeds up or slows down the pattern but retains a metric relationship to the original speed. You get original speed with the 1/16 setting.
Run	Starts and stops the sequencer.

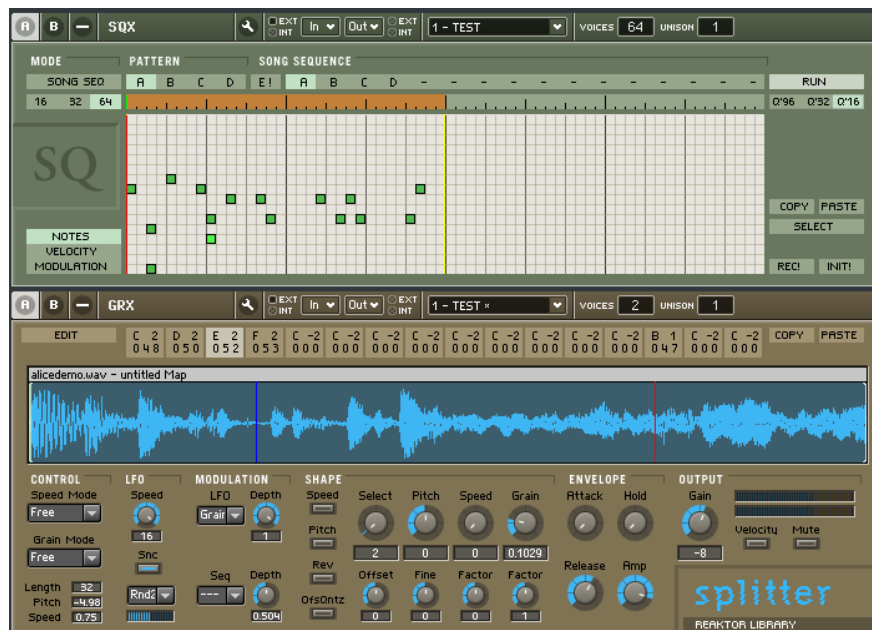
## Sampler

At the heart of the Randomstepshifter lies the sample player. Just load your pre-cut loops into the sample map and the patch will rearrange them. You are free to use the modulation events from the three [Sequencer] tracks to control the parameters for [Sample select], [Sample offset] and [Pitch], or to let these three parameters be randomly varied. Activate random mode (the [Rnd] buttons) and adjust the three [RAND] knobs to get different pseudo-random results. Try moving the [QNTZ] knob in the [Sample offset] module while the sequencer runs for interesting dynamic sample cut-up in real-time. Please make sure that all samples in the sample map have a transposition value of 0.

Sample select	Rnd	This knob selects one of the pseudo-random sequences. Each incoming value from the modulation track is transformed in a pseudo-random way.
	Seq / Rnd / Off	These three buttons switch the modulation modes for the [Select] parameter. Choose between direct modulation by the modulation track, random variation based on the modulation track, and no modulation.
	Select	Sets the base for sample selection via the modulation track. This is the sample that gets played when [MOD] is inactive.
Sample offset	FIRST	Defines the start point of the range for sample selection in the sample map.
	LAST	Sets the end point of the range for sample selection in the sample map.
	RAND	This knob selects one of the pseudo-random sequences. Each incoming value from the modulation track is transformed in pseudo-random way.
	Seq / Rnd / Off	These three buttons switch the modulation modes for the [Offset] parameter. Choose between direct modulation by the modulation track, random variation based on the modulation track, and no modulation.
	MOD	Switches the modulation input from the sequencer on for the [Sample offset] module.
	Offset	Sets the base offset in the sample. This is the offset that is applied when [MOD] is inactive.

	QNTZ	Controls sample offset quantization. 1 = 1/16th, 2= 1/8, 4 = 1/4, etc.
	Smth	Controls for re-synthesis smoothness of sample playback. This alters the sound when introducing extreme pitch settings.
Pitch	RAND	This knob selects one of the pseudo-random sequences. Each incoming value from the modulation track is transformed in a pseudo-random way.
	Seq / Rnd / Off	These three buttons switch the modulation modes for the [Pitch] parameter. Choose between direct modulation by the modulation track, random variation based on the modulation track, and no modulation.
	MOD	Switches on the sequencer's modulation for the [Pitch] module. Sets the transposition of the sample. This is the transposition applied when [MOD] is inactive. This interacts with the [RANGE] control and is independent from tempo as long as [Fit] is inactive.
	Pitch	Sets the range of bipolar sample transposition in semitones. A value of 12 gives you a transposition range from -12 to +12 semitones.
	RANGE	When active, the pitch follows the tempo changes of sample playback, like in a conventional sample player.
Env	Fit	
	Attack	Sets the attack time of an ADSR envelope triggered by the sequencer events.
	Decay	Sets the decay time of an ADSR envelope triggered by the sequencer events.
	Sustain	Sets the maximum level the envelope will reach.
	Release	Sets the time that passes until the envelope is completely faded out after it has reached the sustain level.
	Sel / Offs / P	Selects the trigger input for the envelope. It can be the trigger track from the select-, offset-, or pitch track, respectively.
	on	Switches the envelope on or off.
Output	Mute	Mutes sound output from the sample player.
	Gain	Controls the main volume of the sample player.

# Splitter



The Splitter is a small but sonically flexible sequenced sample-player. Geared towards granular beat production, it can also be used for melodies or padwork. The main idea behind this sequencer / sample-player combo are the 16 sample slots. You can assign different fragments of the selected sample with individual settings for all parameters to the different slots above the waveform display. You can also assign individual MIDI notes.

## Sequencer

The sequencer delivers classic step-sequencing in a very useable package. It offers 16 notes tracks with velocity control plus an additional modulation track, a song mode, and the ability to record incoming MIDI notes. The 16 sample slots (see description of the Splitter below) are represented by the sequencer's 16 notes tracks. The leftmost sample slot corresponds to the bottom track, the rightmost sample slot corresponds to the top track.

Mode	Song Seq	Toggles song mode on and off. If on, the pattern sequence defined under [Song Sequence] is played. If off, the currently selected pattern is played and looped.
	Zoom Level	Chooses whether 16, 32, or 64 steps are displayed. This has no influence on the notes played.
	Notes	Displays the sequencer's notes track. Click into the notes grid to create notes, right-click to delete notes (ctrl-click for Mac users). Note length depends on the quantization settings on the top right side of the sequencer.
	Velocity	Displays the sequencer's velocity track. Every note in the notes grid has a velocity bar. Drag with the mouse to change the levels.
	Modulation	Displays the sequencer's modulation track. Enter the desired modulation steps by dragging with the mouse. Quantized in 16ths.
Song Sequence	A/B/C/D	When not in song mode (see [Song Seq]), the selected pattern is played and looped.
	Song Edit	The edit button enables you to assign patterns to the [Pattern Slots].
Global controls	Pattern Slots	When [Song Edit] is active, click on a pattern slot and drag the mouse up / down to select the desired pattern.
	Loop Bar	The brown bar above the sequencer grid represents the loop region of the sequencer. Right click to set length (ctrl-click for Mac users), left click and drag to move.
	Run	Switches sequence playback on or off.
	Q'96 / Q'32 / Q'16	Quantization setting for the note-length resolution. Q'96 means 96th resolution, Q'32 is 32th resolution, and Q'16 is 16th resolution.
	Copy	Copies the currently selected notes or modulation events, respectively, into the clipboard.
	Paste	Pastes the pattern in the clipboard onto the current pattern.
	Select	Toggles select mode on or off. When on, you can select multiple notes in the notes track via click or by dragging a square around them. You can also select a range in the modulation track.
	Rec !	Switches on note recording via MIDI in.
	Init !	Deletes all notes of the pattern and resets the modulation track events to zero. (You need to double click it!)

## Splitter

The granular sample-player enables you to load samples and trigger defined parts of them with individually stored settings for envelope, pitch, speed and grain length via the built-in sequencer or MIDI input. You also have a tempo-syncable LFO and some control over routing modulations and quantizing sample playback parameters.

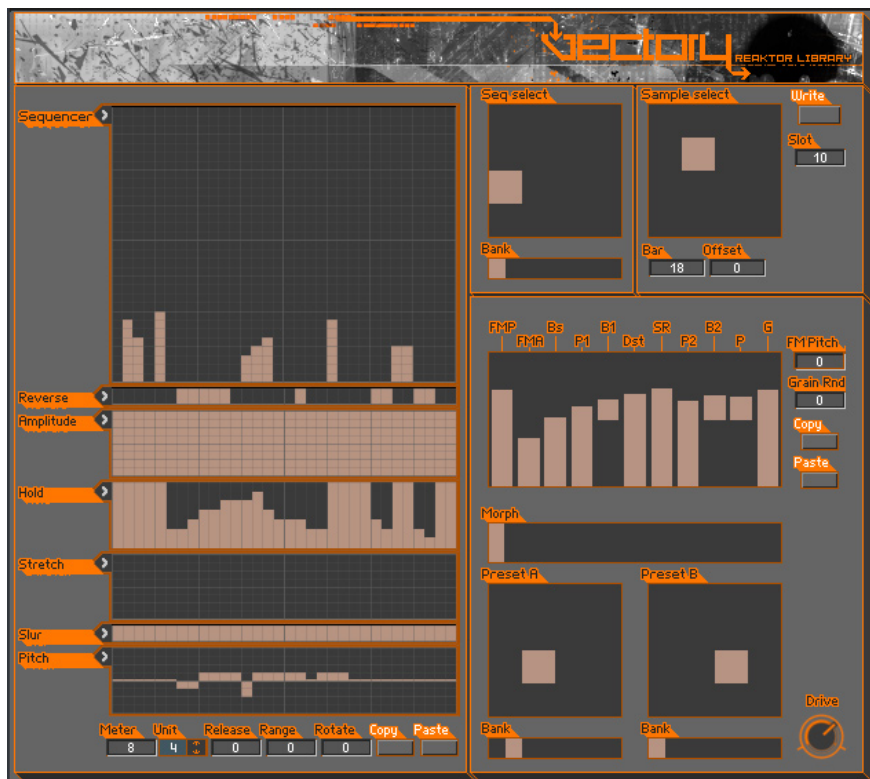
Slots	Edit	If active, you can change the parameters of the selected slot (see [LFO], [Modulation], [Shape], [Envelope] and [Output]). Also, an incoming MIDI note value is assigned to the selected slot. You can also assign notes with the mouse (see [Slots]).
	Slots	If [Edit] is active, you can change the slot's parameters and change the assigned MIDI note by clicking into the slot and dragging up or down. You can also assign MIDI notes via MIDI in (see [Edit])
	Copy	Copies all parameters of the selected slot.
	Paste	Pastes the slot clipboard into the selected slot, overwriting all the slot's parameters.
Control	Waveform Display	Right-click on the title bar of the waveform display to access the sample map menu (ctrl-click for Mac users). When in Edit mode, you can set the red sample-start line by clicking on the waveform.
	Speed Mode	These settings affect the [Speed] parameter of all sample slots. In free mode the speed can be dialed in freely; in grid mode it is quantized to 16th note values.
	Grain Mode	These settings affect the [Grain] parameter of all sample slots. In free mode the grain length can be dialed in freely; in grid mode it is quantized to 16th note values; in note mode the grain length is quantized to steps corresponding to the 127 MIDI notes.
LFO	Length	Shows the length (in 16ths) of the sample currently selected from the sample map.
	Pitch	Shows the sample's pitch deviation from original pitch when played at the current tempo.
	Speed	Shows the sample's pitch deviation when played at the current tempo. 1 represents original speed.
	Speed snc	Adjusts the frequency of the Low Frequency Oscillator. Activates synchronization of the LFO to the song tempo.
	Waveform	With this menu you can choose between six different LFO waveforms (sine, sawtooth, reverse sawtooth, pulse, and two random modes).



Modulations	LFO	Routes the LFO to different parameters. Choose which parameter of the sample-player gets modulated by the LFO's oscillations. Target parameters are [Offset], [Pitch], [Speed], and [Grain].
	LFO Dpth	This control determines how much the LFO affects the chosen parameter.
	Seq	Routes the modulation track of the sequencer to different parameters. Choose which parameter of the sample-player is modulated by the Modulation track. Target parameters are [Offset], [Pitch], [Speed], and [Grain].
	Seq Dpth	This parameter determines how much the modulation track of the sequencer affects the target parameter.
Shape	Sample start point (waveform display)	You set the start point of the sample fragment by clicking into the waveform display. It is indicated by a red line.
	Speed Switch	Binds the sample's playback speed to the song tempo. The effect is similar to speeding up or slowing down a record player.
	Pitch Switch	Binds the sample's playback pitch to the song tempo. The effect is similar to speeding up or slowing down a record player.
	Rev	When active, the sample-fragment is played in reverse, starting from the set start point.
	OfsQntz	When active, the [Offset] parameter works quantized to 16ths.
	Select	Selects a sample from the sample map.
	-offs+	This dial offsets the sample start of the currently edited sample-fragment.
	Pitch	Transposes the pitch of sample playback independently from speed. 0 is original pitch.
	-fine+	Fine-tunes the sample-fragment transposition.
	Speed Knob	Sets the speed of sample playback independently from pitch. 1 is original tempo, 2 is double speed. This interacts with the [Speed Factor] control.
	Speed Factor	With this dial you can multiply the [Speed] control value by a factor. Use it to change the range of the [Speed] knob.
	Grain	Control of grain length. This interacts with the [Grain Factor] control.
	Grain Factor	With this dial you can multiply the [Grain] control value by a factor. Use it to change the range of the [Grain] knob.

Envelope	Attack	Sets the time that passes until the amplitude envelope reaches its peak after triggering.
	Hold	Sets the time the envelope stays at its peak value before it moves to the release phase.
	Release	Sets the time that passes until the amplitude envelope is completely faded out after the hold period has ended.
	Amp	Sets the maximum level the envelope will reach. This gets modulated by velocity if [VelSns] is on.
Output	Gain	Sets Splitter's master volume.
	Mute	Mutes Splitter's audio output.

# Vectory



Vectory is an aggressive sample destruction unit. It consists of a sampler (on the left side) with vast re-arranging capacities whose signal is fed into a grain multi-effect (on the right side) that re-synthesizes the sound.

This structure is optimized for live use, with low-level REAKTOR Core DSP. Complete settings for sample loops, re-arrangements, and grain effects can be recalled by moving the markers within the large square selection displays – changes occur instantly and with no audio drop-out. The effect unit even offers morphing between two settings.

The sample is loaded in a sub-instrument of Vectory called **Sample Loader**. Press Ctrl+2 to open its panel; to return to the main Vectory display press Ctrl+1. Only one sample can be loaded at a time. However, this sample can be quite long and contain several discrete loops.

The second panel set also contains a further sub-instrument labeled **Controllers**. This is designed to automate Vectory's parameters via MIDI / VST.

## Sample

This section at the top-right of the panel selects the sample material from the Sample Loader. By using the large square markers one of sixteen slots can be selected; each slot contains the data of the sample loop's beginning, measured from the start of the sample file in bars and sixteenth.

Sample Selection Display	Selects the active sample loop slot; each slot stores independent values for [Bar] and [Offset]. Those two parameters control the starting point within the loaded sample; thus, they define the sample material being played, which is then subject to re-arrangement by the [Sequencer] section. The length of the loop within the sample is controlled by [Sequencer][Meter] and [Sequencer][Unit].
Write Slot	Stores the current values of [Bar] and [Offset] into the current slot. Displays the number of the active slot in the [Selection Display].
Bar	Sets the starting point of the sample readout. This control adjusts the number of bars to be skipped in the sample file. (See also [Sample Loader][Bar] and [Sequencer][Position].)
Offset	Sets the starting point of the sample readout. This control adjusts the number of sixteenth steps that are added to the number of bars set by the [Bar] control. (See also [Sample Loader][Tempo] and [Sequencer][Position].)

## Sequencer

The sequencer contains two sections: The [Seq Select] part chooses one of several sequencer settings; each setting is defined within the main [Sequencer] part that fills the left side of the instrument's panel. The sequencer pattern is mapped onto the sample material selected in the [Sample] section.

Sequence Selection Display	Selects the active sequencer pattern. There are sixteen slots in each bank. (See also [Bank].)
Bank	Selects the bank from which the [Selection Display] loads the sequencer pattern. There are eight banks available.
Position	Defines the rearrangement pattern. The sequence is read in sixteenth steps from left to right; the vertical axis sets the offset from the sample readout starting point for each step in sixteenths; e.g. a scale from the bottom-left to the top-right represents the normal sample readout while a scale from the bottom-right to the top-left results in inverse readout: first the last sixteenth of the sample, then the one before the last etc. The starting point of the sample readout is controlled within the [Sample] section.
Reverse	Defines whether the sixteenth note selected by the [Position] pattern is played from the end to its beginning or in its normal playback direction.
Amplitude	Adjusts the amplitude for each sequencer step.
Hold	Adjusts the hold time for each sequencer step. (See also [Release].)
Stretch	Lengthens the sample at this sequencer step. The higher the value, the more it is stretched – the first square represents a ratio of 2:1, the next one is 3:1, etc. The sample is stretched by a grain re-synthesis algorithm; therefore, the grain pitch and grain frequency parameters of the [Grain Effect][Parameter Display] will greatly affect the sound of stretched steps. As the sequence moves on without being influenced by the stretch, parts of the stretched sample that do not fit in the step are cut (at a ratio of 2:1 the second half will be cut, etc.) See also [Slur].
Slur	Ties stretches over consecutive sequencer steps. If slur is off, the stretch will be re-triggered at each sequencer step; if it is switched on, the stretch will be continued. This also affects the [Reverse] function.
Pitch	Adjusts the pitch shift for each sequencer step. The values set here are relative ones; the absolute range of shifting is controlled by [Range].
Meter	Controls the loop length in steps; the step length is adjusted by [Unit].
Unit	Sets the rhythmic unit (fourth, eighth or sixteenth, according to the current MIDI tempo) used as step for the [Meter] control.

Release	Adjusts the release time after each sequencer step's hold period. (See also [Hold].)
Range	Sets the absolute pitch range available for the sequencer steps. To actually adjust the pitch in each step, use the [Pitch] pattern, which exerts relative control over pitch. With higher [Range] values, identical [Pitch] patterns produce more drastic pitch shifts.
Rotate	Sets an offset to the sequencer readout.
Copy	Copies the current sequencer pattern into a buffer that can be read out by pushing the [Paste] button. A complete pattern can easily be duplicated or moved by selecting another slot with [Sequencer Select][Selection Display] and [Sequencer Select][Bank] before pasting the buffer.
Paste	Pastes the buffer's data into the current sequencer pattern, overwriting the old values. (See also [Copy].)

## Grain Effect

This section controls the multi-effect placed after the sampler and the re-arrangement sequencer. The separation between sound generator and effect unit is only true for the panel; internally, those sections are closely interrelated. For example, both the frequency modulation and the grain resynthesis parameters show no results within the effect unit, but inside the sampler itself; they are placed here because they impact the instrument's sound as much as the other effect parameters.

There are two slots named A and B containing two different sets of effect unit settings. The [Morph] control interpolates between both settings for smooth transitions in live use.

Parameter Display	Shows the currently active effect parameters. If [Morph] is set completely to the left or right – thus selecting either the A settings or the B settings and not an interpolation – the parameters can also be edited. There are eleven parameters: FM Pitch, FM Amount, Bias, Pre-Quantize EQ Frequency, Pre-Quantize EQ Amount, Distortion (overdrive saturation), Sample Rate Reduction (frequency quantization), Post-Quantize EQ Frequency, Post-Quantize EQ Amount, Grain Pitch, Grain Frequency. Their technical meaning cannot be explained here in detail; their influence on the sound, however, can easily be heard when changing the values.
Grain Random	Adjusts the amount of randomness applied to the grain synthesis. The lower the value, the more constant is the frequency at which new grains are generated.
Copy	Copies the current effect parameter setting into a buffer that can be read out with the [Paste] button. The data can easily be moved to another storage position by selecting another parameter slot with [Morph], [A/B Selection Display] and [A/B Bank] before pasting.

Paste	Copies the buffer's data into the current parameter setting, overwriting old values. (See also [Copy].)
Morph	Interpolates between the parameter settings selected with [A Selection Display] and [B Selection Display]. Move the marker completely to the left to activate and edit preset A; move it completely to the right to activate and edit preset B. Moving the marker in the space between gradually morphs from one preset to the other; during morphing no editing of the preset settings is possible.
Preset A	Selects the slot whose parameter settings are active (and editable if [Morph] is moved completely to the left). Each bank has sixteen slots. (See [A Bank].)
Bank A	Selects the bank from which the [A Selection Display] loads its data. Eight banks are available.
Preset B	Selects the slot whose parameter settings are active (and editable if [Morph] is moved completely to the right). Each bank has sixteen slots. (See [B Bank].)
Bank B	Selects the bank from which the [B Selection Display] loads its data. Eight banks are available.
Drive	Adjusts the amount of compression applied to the final output signal. High values represent high compressor thresholds; all audio data below this threshold will be amplified.

## Sample Loader

The [Sample Loader] imports audio material. Only one sample can be loaded, but you can assign playback for different loops and parts.

BPM	Sets the tempo of the loaded sample in beats per minute. This should be done accurately – three small boxes to the right of the main BPM box allow you to set the tempo with three decimals – as this value is used to calculate the positions within the sample file (see [Sample][Bar] and [Sample][Offset]).
Start	Adjusts an offset in milliseconds at the beginning of the sample file that is skipped by all calculations concerning positions within the sample file.
Bar	Sets the number of sixteenth notes (according to the tempo adjusted in [BPM]) within one bar (see [Sample][Bar]).

## MIDI Controller

This sub-instrument of Vectory provides various automation possibilities to control the parameters via MIDI or VST. Five MIDI continuous controllers can be selected as modulation sources [Control A] to [Control E]. Additionally, two two-dimensional sources are available as [XY1] and [XY2]; they are controlled by two MIDI CCs, one for the horizontal movements, the other one for vertical ones. [XY2] can also be controller via the MIDI pitch. Those modulation sources can be assigned to various parameters of Vectory within the [Assignment] section of the sub-instrument.

Control A .. E		Selects the MIDI CC number that is referred to as [Control A] to [Control E] respectively.
XY1	X	Selects the MIDI CC number that controls the horizontal position of the marker.
	Y	Selects the MIDI CC number that controls the vertical position of the marker.
XY2	X	Selects the MIDI CC number that controls the horizontal position of the marker.
	Y	Selects the MIDI CC number that controls the vertical position of the marker.
	Note	Switches between MIDI CC mode (off) and MIDI note mode (on). In MIDI note mode, the position of the marker is controlled by the pitch of incoming MIDI events. The pitch adjusted by [Origin] selects the first position, the next pitch selects the second position etc.
	Origin	Sets the MIDI pitch that selects the first position of the marker if [Note] is on.
Assignments		The four square displays [Sample / Sequence / A / B Selection Display] of the main instrument can be controlled by all seven modulation sources; all other parameters only provide the five one-dimensional sources [Control A] to [Control B].



# Effects

## FlatBlaster 2



The great final-stage mastering tool FlatBlaster 2 has been rebuilt using the new REAKTOR Core features. This patch combines four frequency-specific compressors with a full-spectrum peak-limiter to produce a high-end package for your multiband dynamics shaping needs. As it does not introduce any delay it is not limited to mastering use but can also be applied on a per channel basis. The controls might appear intimidating at first glance, but are actually straightforward when you examine the signal chain. The separately compressed bands get mixed together and then processed by a full-band peak limiter. Please note that the master bypass for the complete patch is situated to the left above the X Over section.

### Multi-band compressor

After the input stage the signal gets split into four independent frequency bands as defined by the X Over section. Each frequency band gets processed by independent, identical compressors and can be independently muted, soloed and bypassed. Separate saturators for each band make it possible, for example, to add punch and bite to the mids without affecting clarity in the lower registers.

Input	Input	Trims the input gain to prevent overload.
	Bypass	Bypasses the complete effect. This is the master bypass switching off all compressors and the Limiter.
X Over	High	Sets the crossover frequency between the High and Mid High compressor bands.
	Mid	Sets the crossover frequency between the Mid High and Mid Low compressor bands.
	Low	Sets the crossover frequency between the Mid Low and Low compressor bands.
High, Mid High, Mid Low and Low Compressors	Stereo	Sets stereo width of the frequency band. 0 is mono, 1 is original stereo, 2 is extra stereo.
	Tresh	Sets the point at which the compressor will begin to work (in db). Levels below this threshold remain unprocessed.
	Ratio	Adjusts the ratio of the input level to the output level after compression.
	Knee	This parameter adjusts how gradually the full amount of compression is introduced. Think of it as a slope control for the attack time.
	Sat	Drives the band into saturation.
	Link	Activates stereo linking of the two input channels. When active, the compressor takes the max of the left and right peak levels and uses it for both channels. This preserves a clean stereo image and is lighter on CPU cycles.
	Att	This dial adjusts the attack time. It is the time the compressor takes to react to an above-threshold signal.
	Rel	With this control you set the release time. This is the time the compressor takes to return the signal to normal when it falls below the compression threshold.
	Out Gain	Sets the amount of amplification applied to the compressed signal of the specific band before it gets mixed with the other bands.
	Bypass	Bypasses the compressor for the respective band.
	Mute	Turns the sound of the respective band off.
	Solo	Turns all other bands off, leaving only the signal of the soloed band. Use it to fine tune single compressor bands.

## Full-band peak limiter

The peak limiter affects the full band signal. For clean mastering purposes we recommend a limiter threshold setting of about -3 to -4 db and a peak setting at 0db. Should pumping effects be desired, adjust the threshold to more extreme values.

Thr	Adjusts the threshold of the limiter. Levels above this value get processed.
Peak	Adjusts the hard limit of the signal. No signal will exceed this limit.
Rel	This adjusts the release time. It is the time the limiter takes to return the signal to normal when it falls below the limiting threshold.
Soft / Hard	Balances between soft saturation and hard clipping of the above peak signal.
Compare	Controls the amplification of the uncompressed signal if Bypass is active. If you want compression without amplification set it to 0 and make sure there is no change in level when toggling the Bypass button.
Link	Activates stereo linking of the two input channels.
Bypass	Bypasses the Full Band Peak Limiter only, leaving the 4 compressors active.

# Lurker



Lurker is a hybrid effect capable of classic phaser sounds, spring reverbs and feedback echoes – but most of all it transforms any incoming signal into stunning rhythmic sequences, mangling pitches and re-arranging the sound. This is technically possible because all those effects are based on a delay unit (and this instrument is an extremely versatile one).

Four internal sequencer tracks are the most prominent feature. They allow for fast, visual creation of musical patterns that you can use to modulate parameters such as delay times of the two independent delay units. Those times can be set in sixteenth note multiples (for tempo-based effects) or in milliseconds (for comb filter-like effects that map a new pitch onto the signal). A filter, a gating envelope generator, and a final delay further enrich the sound.

## Global

This top section of the instrument panel contains three parts: the input control (at the left), the snapshot management (in the middle), and the shuffle control (at the right).

The input control provides a simple sampler to load files and re-trigger their playback synchronized to the sequencers. The level of external signals can be controlled here. The snapshot handling and the shuffle system are identical to those of Massive; see that instrument's manual for further details.

Input	Loop Switch	Controls the events that re-trigger the sampler. If on, the sampler starts playback at the file's beginning when the loop controlled by [Length Control] and [Unit Select] is returning to its origin; if off, the sampler is re-triggered only when the global MIDI clock starts playing.
	Length Control	Sets the length of the loop that controls the sampler's re-triggering if the [Loop Switch] is on. (See also [Unit Select].)
	Unit Select	Selects the rhythmical unit on which [Length Control] is based. This unit refers to the global MIDI clock.
	Sampler	Displays the currently active sample (see [Sample Select]). Double-click to open the Sample Map Editor where sample files can be loaded and organized.
	Sample Select	Selects one of the samples loaded into the [Sampler].
	Sample Pitch	Transposes the selected sample. This also affects the samples playback speed. (An octave up or down sets the playback to double or half speed respectively.)
	Internal Level	Controls the amplitude of the sampler.
	External Level	Controls the amplitude of the external signal.
	External Mute	Disables the external input.
	External Display	Shows the level of the external signal.
Snapshot	Snapshot Store	With the left mouse button, a snapshot slot number can be selected; by pressing the right mouse button, the current instrument settings (including all sequencer data) are stored into this snapshot slot.
	Snapshot Recall	Displays a list of the available snapshots; selecting a snapshot with the mouse recalls all its data, including sequences.
	Snapshot Mode	Selects whether the snapshots are only recalled internally or if external control signals received at the instrument's [Snap] port are recognized, too. This allows connection to a master song sequencer.

Shuffle	Quantization Select	Selects one of twelve quantization presets. Each preset ranges over sixteen steps; the higher the value within the display, the more delay is applied to this step. The first preset, for example, alternates between low and high values, so every second step will be delayed, resulting in a standard off-beat shuffle. The presets only define relative times; the effective delay time at maximum values is set by the [Shuffle] control.
	Shuffle	Scales the preset of the [Quantization Select] control. Turn to the left for no quantization – independent of the selected preset –, to the right for full delay times.

## Sequencer

There are two step sequencers (tracks [A] and [B]) and two tracks that glide from step to step ([C] and [D]). Each sequencer provides individual control over length and speed.

Length Control	Sets the length of the loop that can be edited within the sequencer display in steps. (See also [Unit Select].)
Unit Select	Selects the rhythmic unit with which each step of the sequencer track is interpreted. This refers to the global MIDI clock.
Sequencer	Defines and displays the track's rhythmic pattern.

## Delay Units

Two identical delay units form Lurker's core. They can be used either in parallel or serially. Each offers independent delay times for the left and the right audio channels, both defined as multiples of sixteenth notes or in milliseconds. At the left of the controls that adjust the delay times, their modulation can be controlled; even the depth of modulation is subject to modulation, resulting in complex interactions of several modulation patterns. The knobs at the delay time controls' right define the channel swap, the amount of feedback and the feedback's filtering.

Depth	Sets the amount of modulation applied to the delay time. This is independent of the static delay time and ranges from no modulation (at the left) to a modulation of about 260 milliseconds (at the right). The modulation signal is selected by the [Modulations Source] control below. (See also [Depth Modulation Amount].)
Modulation Source	Selects the sequencer track that modulates the delay time. The amount of modulation at maximum modulation signals is controlled by [Depth].

Depth Modulation Amount	Adjusts the amount of modulation of the [Depth] control. Turn to the left for inverted modulation (i.e. there is much modulation at low modulation signals and vice versa), to a mid position for no modulation and to the right for normal modulation. High values at the right result in very much modulation of the [Depth] control, increasing its maximal modulation amount to approx. 2400 milliseconds. The signal that actually modulates the [Depth] control is selected below.
Depth Modulation Source	Selects the sequencer track that modulates the modulation depth. The amount of modulation is controlled by [Depth Modulation Amount].
Modulation Slur	Sets the amount of interpolation applied to subsequent steps of the modulation track. Turn to the left for no interpolation and fast delay time changes, to the right for soft and slow ramps between different states.
Modulation Invert	Inverts the modulation signal i.e. if on, the modulation signal is not added to the static delay times controlled by [Quantized Delay Time Left / Right] and [Millisecond Delay Time Left / Right], but subtracted.
Quantized Delay Time Left / Right	Sets the static delay time of the left channel (upper control) and right channel (lower control) respectively in multiples of sixteenth notes of the global MIDI clock. The actual delay time is calculated from the sum of this value, the delay time adjusted by [Millisecond Delay Time Left / Right] and the modulation signal (see [Depth]),
MillisecondDelay Time Left / Right	Sets the static delay time of the left channel (upper control) and right channel (lower control) respectively in milliseconds. The actual delay time is calculated from the sum of this value, the delay time adjusted by [Quantized Delay Time Left / Right] and the modulation signal (see [Depth]),
Channel Swap Amount	Controls the modulation applied to the interaction of the left and right feedback signal. At low modulation signals the left channel's feedback signal is routed again to the left channel; at mid modulation values both channels are mixed to a mono sound that is fed back into both channels identically; at high modulation signals the channels are swapped and the left channel's signal is routed to the right channel (and vice versa). This control scales the modulation signal, i.e. at mid position high modulation signals are mapped to mid modulation signals; at the complete left there is no modulation and no channel swapping. The modulation signal is selected below.
Channel Swap Modulation Source	Selects the sequencer track that modulates the channel swap. The amount of modulation is controlled by [Channel Swap Amount].
Cutoff	Sets the cut-off frequency of the low-pass filter within the feedback loop.
Reset	Sets all controllers of the delay unit to their default values.

Feedback Amount	Controls the amount of feedback.
Bypass Switch	Toggles between the dry, unprocessed signal (when on) and the wet, delayed signal.
Mode Select	Switches between parallel and serial modes. In parallel mode, both delay units receive the same input signal and the [Crossfade] control can crossfade between their output signals. In serial mode, the signal enters the upper delay unit and is then routed to the lower unit.
Crossfade	Mixes between the sound of the upper and lower delay unit when [Mode Select] is set to parallel.

## Filter

The filter is placed after the two delay units. The low-pass filter's cut-off frequency and resonance can be edited (you can adjust left and right channels independently); the cut-off can also be modulated by one of the four modulation tracks.

Cutoff	Adjusts the cut-off frequency of the filter. The horizontal axis controls the left channel, the vertical axis the right one.
Cutoff Modulation Amount	Sets the amount and polarity of the modulation applied to the low-pass filter's cut-off frequency.
Cutoff Modulation Source	Selects the modulation track that is used to modulate the filter's cut-off frequency.
Resonance	Adjusts the resonance of the filter. The horizontal axis controls the left channel, the vertical axis the right one.
Reset	Sets all controllers of the filter to their default values.



# Master and Envelope

The master section simply controls the instrument's output level before its signal passes to the additional delay. The [Env] control enables an envelope generator that is triggered by one of the two step sequencer tracks. This can be used to gate the instrument's signal.

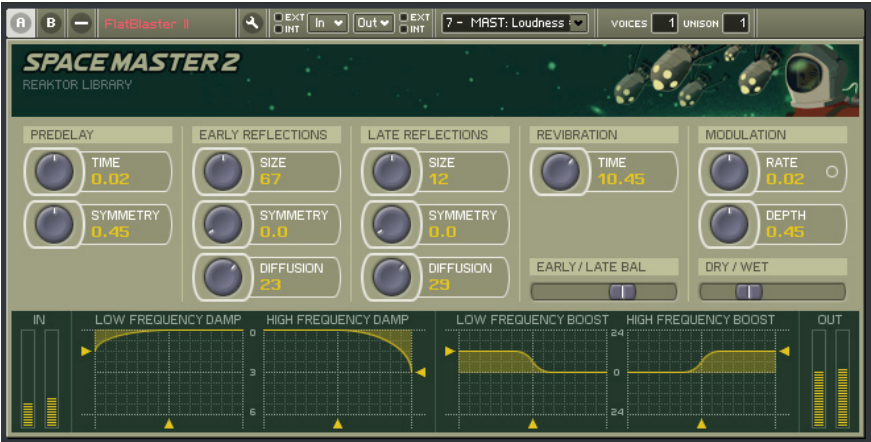
Master	Output	Controls the instrument's main output level.
	Bypass	Mutes the effect and directly routes the input signal to the output.
	Envelope Amount	Adjusts the influence of the envelope generator on the output amplitude. Turn to the left for complete independence. Turn to the right for full shaping of the amplitude by the envelope.
Envelope	Source Select	Selects one of the two step sequencers to be used as trigger signal. (See also [Gate Threshold].)
	Gate Threshold	Controls which steps of the selected modulation track are used as trigger signals. All steps with values below the one adjusted by this control are ignored.
	Velocity Amount	Controls the influence of the trigger gate's height (i.e. velocity) on the envelope's amplitude. Turn to the left for full amplitude with every trigger signal, turn to the right to map the step's value onto the envelope's amplitude.
	Velocity Attack	Controls the amount of modulation applied to the envelope's attack time by the triggering step's velocity. At low velocities (i.e. low step values) the attack time is increased if the knob is turned to the right. At left positions the velocity doesn't affect the attack time. This is independent of the [Velocity] control.
	Velocity Decay	Controls the amount of modulation applied to the envelope's decay time by the triggering step's velocity. At low velocities (i.e. low step values) the decay time is decreased if the knob is turned to the right. At left positions the velocity doesn't affect the decay time. This is independent of the [Velocity] control. (See also [Decay].)
	Decay	Sets the static decay time of the envelope that can be modulated by the triggering step's velocity (see [Velocity Decay]).

## Additional Delay

The delay unit after the output section allows for further manipulation of the signal. It is similar to the main delay units, but the delay times can't be modulated and the channels can't be swapped; instead, there is a high-pass filter within the feedback loop. As a special feature, the ratio between dry, unprocessed signal and wet, delayed sound can be modulated by one of the modulation tracks.

Quantized Delay Time Left / Right	Sets the static delay time of the left channel (upper control) and right channel (lower control) respectively in sixteenth note multiples of the global MIDI clock. The actual delay time is calculated as the sum of this value and the delay time adjusted by [Millisecond Delay Time Left / Right].
Millisecond Delay Time Left / Right	Sets the static delay time of the left channel (upper control) and right channel (lower control) respectively in milliseconds. The actual delay time is calculated as the sum of this value and the delay time adjusted by [Quantized Delay Time Left / Right].
Feedback Amount	Controls the amount of feedback.
Highpass	Sets the cut-off frequency of the high-pass filter within the feedback loop.
Lowpass	Sets the cut-off frequency of the low-pass filter within the feedback loop.
Mix Modulation Source	Selects the sequencer track modulating the ratio between dry, unprocessed signal and wet, delayed sound (see [Mix]).
Mix	Controls whether the ratio between dry, unprocessed signal (heard at mid position) and wet, delayed sound (at the right). Turn to the left to use the sequencer track selected by [Mix Modulation Source] as control: At high modulation levels the wet sound is passed on; at low values, the dry signal.

# Space Master 2



The well-known Space Master series of reverb modellers has been updated for REAKTOR 5. Based on several diffusion delays, Space Master 2 can produce a wide array of high-quality natural or experimental ambiances. The patch's efficient set of reverb parameters include an early reflections section, a late reflections module and a post EQ. Dials for main reverb time, control of balance between the two reflection stages, and between dry and wet signal round off the controls.

## Input and output stage

You can introduce an initial delay into the reverb signal with the predelay [Time] dial and control the predelay's stereo position with the [Symmetry] knob. The [Early / Late Balance] slider can be used to move the source in space – more early reflections bring the signal to the front and more late reflections make it appear further back in space. At the end of the signal chain, the [Dry / Wet] slider crossfades between the dry original signal and the processed sound.

Predelay	Time	Sets an initial delay for the wet signal.
	Symmetry	Introduces a difference into the delay times for the right and left predelay channels. Use this to shift the signal around in the stereo image.
Mixing	Early/Late Balance	With this parameter you can set how much of the early and late reflections, respectively, can be heard in the output.
	Dry / Wet	This controls the balance between dry and wet signal.

## Reflections

Use the two [Size] and [Diffusion] parameters to dial in the early and late stages of variable density diffused reflections. The early stage commonly represents the direct response of the virtual space, whereas the late reflections define the sound when the early reflections have died away.

For dynamic reverb effects you can use the Modulation section. It offers an LFO routed to the delay times with [Rate] and [Depth] control. The LFO can enhance your reverb signal by adding liveliness.

Early / Late Reflections	Size	Determines the range of space generated by the early or late reflections modules by adjusting delay time of the underlying diffusion delays. Higher values give the impression of larger spaces.
	Symmetry	Introduces a stereo shift into the generated reflections.
	Diffusion	Adjusts the perceived density of the generated reflections. Dial for a sparser or fuller reverb sound.
Modulation	Reverberation Time	This control alters the decay time of the reverb response.
	Rate	Control of LFO frequency modulating the delay times.
	Depth	This adjusts the LFO's modulation depth. Higher values give you higher amplitude of the modulation.

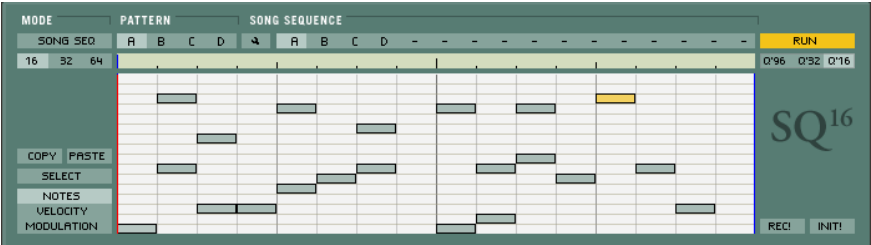
## Frequency response

The two EQ sections serve slightly different needs. The Damping EQs are integrated into the reflection stages and influence their frequency responses. The Post EQ acts on the main output of the patch should be used to color the overall sound.

Frequency Damping	Low Frequency Damp	Low shelving filter that cuts into diffusion delay frequency response of both early and late reflections. Use the horizontal slider to adjust cutoff frequency and the vertical slider to adjust cut or boost.
	High Frequency Damp	High shelving filter that cuts into diffusion delay frequency response of both early and late reflections. Use the horizontal slider to adjust cutoff frequency and the vertical slider to adjust cut or boost.
Post EQ	Low Frequency Boost	A low shelving EQ that acts on the main output of the reverb. Use the horizontal slider to adjust cutoff frequency. The vertical slider adjusts cut or boost.
	High Frequency Boost	A high shelving EQ that acts on the main output of the reverb. Use the horizontal slider to adjust cutoff frequency. The vertical slider adjusts cut or boost.

# Sequencer

## SQ16



### Description

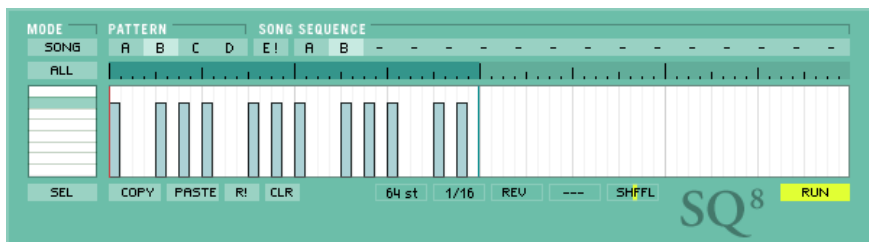
The SQ16 sequencer delivers classic step-sequencing in a very useable package. It features 16 notes tracks with velocity control plus additional modulation tracks, a song mode, and the ability to record incoming MIDI notes.

### Details

Control	Song Seq	Toggles song mode on and off. When on, the pattern defined under [Song Sequence] is played. When off, the currently selected pattern is played and looped.
	Zoom Level	Choose whether 16, 32, or 64 steps are displayed. This has no influence on the notes played.
	Notes	Displays the notes track of the sequencer. Click the notes grid to create notes, right-click to delete notes (ctrl-click for Mac users). Note length depends on the quantization settings on the top right side of the sequencer.
	Velocity	Displays the sequencer's velocity track. Each note in the notes grid has a velocity bar in the velocity track. Drag with the mouse to change the levels.
	Modulation	Displays the sequencer's modulation track. Enter the desired modulation steps by dragging with the mouse. Quantized in 16ths.
Pattern	A/B/C/D	When not in song mode (see [Song Seq]), the selected pattern is played and looped.

Song Sequence	Song Edit	The edit button enables you to assign patterns to the [Pattern Slots].
	Pattern Slots	When [Song Edit] is active, click into a pattern slot and drag the mouse up / down to select the desired pattern.
Global controls	Loop Bar	The brown bar above the sequencer grid represents the loop region. Right-click (ctrl-click for Mac users) to set length and left-click and drag to move.
	Run	Switches sequence playback on or off.
	Q'96 / Q'32 / Q'16	Quantization setting for the note-length resolution. Q'96 means 96th resolution, Q'32 is 32th resolution, and Q'16 is 16th resolution.
	Copy	Copies the currently selected notes or modulation events into the clipboard.
	Paste	Pastes the pattern clipboard into the current pattern.
	Select	Toggles select mode on or off. When on, you can select multiple notes in the notes track by clicking or by dragging a square around them. You can also select a range of the modulation track.
	Rec ! Init !	Switches on note recording via MIDI in. Deletes all pattern notes and resets the modulation track events to zero. (You need to double click on it!)

## SQ8



### Description

The SQ8 is your standard building block for rhythmic step-sequencing. It sports a clean interface: 4 patterns with 8 tracks (consisting of 64 steps each). You also get variable looping, shuffle, reverse play, and multiple viewing options. On top of that, you can chain 16 patterns together into a song.

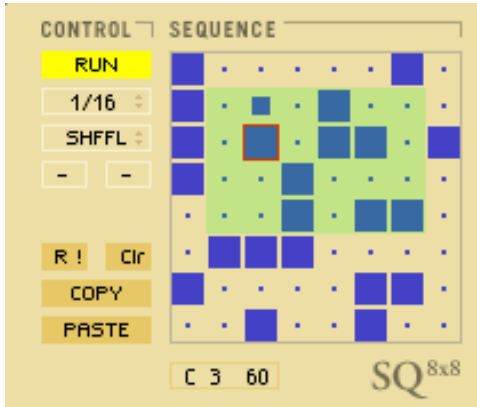
## Details

Mode	Song Seq	Toggles the song mode on and off. If on, the pattern sequence defined under [Song Sequence] is played. If off, the currently selected pattern is played and looped.
	Pattern	When not in song mode (see [Song Seq]), the selected pattern is played and looped.
Song Sequence	Song Edit	The edit button enables you to assign patterns to the [Pattern Slots].
	Pattern Slots	When [Song Edit] is active, click into a pattern slot and drag the mouse up / down to select the desired pattern.

Global controls	Pattern view / Track view	Click on [All] to see the complete pattern with all tracks. Click on any button to the left of a track to view the track exclusively. In track view you can also change the velocity of the individual notes.
	Loop Bar	The darkish green bar above the sequencer grid represents the sequencer's loop region. Right-click (ctrl-click for Mac users) to set length and left-click and drag to move.
	Notegrid	Click into the grid to add or delete note events.
	Sel	Toggles note select mode on or off. When on, you can select an area of the note grid to be cleared, copied from, or pasted to. This works in all viewing modes.
	Copy	Copies the content of the current pattern.
	Paste	Pastes the pattern clipboard into the current pattern, overwriting all events.
	Rec !	Switches on note recording via MIDI in.
	Clr	Deletes all selected notes of the pattern.
	Zoom Level (16 st, 32 st, 64 st)	Click and drag mouse up or down to zoom in and out of the currently displayed pattern.
	Clock divider (1/6, 1/8, 1/12, 1/16, 1/24, 1/32)	Chooses between different clock divisions. This speeds up or slows down the pattern but retains a metric relationship to the original speed. You get original speed with the 1/16 setting.
	Rev	Toggles reverse play on and off. The direction is reversed by pattern mirroring.
	Stepshifter	This menu determines the playback mode. --- is normal, 1324 and 1432 let the steps swap their positions, <?> plays in random direction, <??> randomly jumps to the previous or next step, ???? jumps to a completely random step.
	Shffl	Shuffle function. Click and drag mouse up or down to select the amount of shuffle.
	Run	Starts and stops the sequencer.



## SQ 8x8



### Description

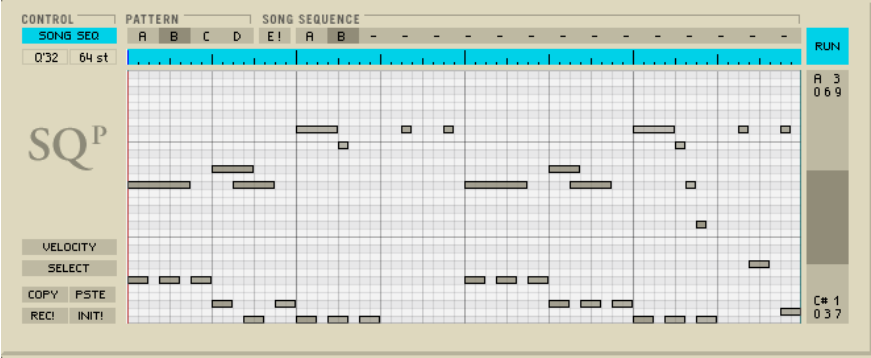
The SQ 8x8 is a small step-sequencer with a twist. You put events in a grid and drag a rectangle around a group of them by right-clicking and dragging (ctrl-click when you're on a Mac). This rectangle defines the sequencer's loop area, controlling what gets played line-wise. You can change this area in realtime. Think of it as a two-dimensional loop-bar. Some nice realtime step-shifting and shuffle features are also part of the package.

### Details

Mute	Mutes the sequencer output.
Notegrid	Click the grid to add or delete note events. Click and drag the mouse up or down to change velocity. Right-click (ctrl-click for Mac users) and drag to define the loop-area.
Clock divider (1/6, 1/8, 1/12, 1/16, 1/24, 1/32)	Choose between different clock divisions. This speeds up or slows down the pattern but retains a metric relationship to the original speed. You get original speed with the 1/16 setting.
Shffl	Shuffle function. Click and drag mouse up or down to adjust the amount of shuffle.
X - playback modes	-- - walk normally X - +/- random step in X direction XX - +/- random step in whole X row

Y - playback modes	-- - walk normally Y - +/- random step in Y direction YY - random in whole column
R!	Randomizes the current loop area.
Clr	Clears the current loop area.
Copy	Copies the content of the current loop area.
Paste	Pastes the pattern clipboard into the current loop area, overwriting all events.

# SQP



## Description

The SQP is a piano roll-style sequencer covering a very wide midi note-range. You can enter notes via the mouse or record incoming MIDI notes. If you want to input longer events with the mouse, just click and drag the start or end of an existing note. Move events by clicking and dragging them around. When [Select] is on you can move selected events as a group.

## Details

Control	Song Seq	Toggles song mode on and off. If on, the pattern sequence defined under [Song Sequence] is played. If off, the currently selected pattern is played and looped.
	Quantization	Controls the quantization of note events. Choose between quantization in 16th, 24th and 32th. You can also switch off quantization.
	Zoom Level	Choose whether 16, 32, or 64 steps are displayed. This has no influence on the notes played.
	Velocity	Displays the sequencer's velocity track. Each note in the notes grid has a velocity bar in the velocity track. Drag with the mouse to change the levels.
	Select	Toggles select mode on and off. If on, you can select note events by clicking them or by dragging a square around them.
	Copy	Copies the currently selected events.
	Pste	Pastes the pattern clipboard into the current pattern.
	Rec !	Switches on note recording via MIDI in.
Song Sequence	Init !	Deletes all note events of the pattern. (Needs to be double clicked!)
	A/B/C/D	When not in song mode (see [Song Seq]), the selected pattern is played and looped.
	Song Edit	The edit button enables you to assign patterns to the [Pattern Slots].
Global controls	Pattern Slots	When [Song Edit] is active, click into a pattern slot and drag the mouse up or down to select the desired pattern.
	Loop Bar	The blue bar above the sequencer grid represents the sequencer's loop region. Right- click (ctrl-click for Mac users) to set length and left-click and drag to move.
	Run	Starts and stops the sequencer.
	Roll bar	To the right of the note grid you will find the roll bar that navigates the MIDI note range. Drag it up or down to see the higher or lower registers, respectively.

# Index

## Symbols

? Menu .....	72
1/96 Clock .....	93
16-Step Sequencer Plus Bassline	31
2D Table	
Default.....	106
Max.....	106
Min .....	106
Show Event Initialization Order	. 60

Automatic	
Layout .....	57
Panel Layout .....	91
Voice Reduction .....	63
Automation ID editing.....	15
Auto Save Configuration.....	16
Auto Voice Reduction .....	101
Available in Panelsets.....	106

## A

About .....	72
Accessing Files.....	78
Adding Instruments.....	94
Adding Modules.....	121
Akai Import .....	194
Alpha Channel .....	165
Always on top .....	61
Animation .....	164, 165
Height .....	165
Width.....	165
Appearance Page .....	90, 105, 118
Append snapshot .....	171
Arrange Icons .....	71
Arranging the Panel.....	43
Audio + MIDI Settings .....	60
Audio-In Module .....	27, 83
Audio-Out Module.....	27, 84
Audio files path .....	65
Audio In level meter .....	74
Audio Out level meter .....	74
Audio signals.....	135
Auditioning Files .....	80
Auditioning Sample Files.....	192

## B

Background	
Bitmap.....	107, 119
Bitmap, Macro .....	119
Backup Data with Module .....	198
Bank .....	174
Basic Operation .....	50
Batch Processing .....	55
Beatloop .....	180
BMP .....	165
Bookmark .....	76
jump to.....	76
BPM by Snapshot .....	88
Browser .....	77
Browser and Snapshots	
always on top .....	61
Button .....	142, 156

## C

Cascade.....	71
Choose Color .....	90, 105
Classic Modular .....	208
Clock	
Start.....	58
Stop .....	58

Close .....	71	Directories .....	64
Close All Structures.....	71	Disable Automation .....	155
Color, edit Instrument ... ..	105	Display .....	209
Colors, edit Ensemble.....	90	DIY.....	33
Color Scheme.....	91, 105	Double-click .....	50
Comparing Snapshots.....	173	Draw/Select/Control Mode.....	204
Connection.....	110, 154	Drop-down Menu .....	157
Connection Page.....	92, 107	Drum-Maps .....	183
Connection Properties Controls.	152	Duplicate .....	56, 96
Constant Sources.....	128		
Cont/Note .....	154	<b>E</b>	
Context Menu.....	51, 95, 146	Editing Panels .....	155
Context Menu, Macro .....	114	Edit Menu .....	55
Context Menu, structure .....	136	Edit Sample List Box.....	188
Control		Ensemble.....	81
ID Numbers.....	169	is not Found .....	17
Rate .....	57, 89	Panel Toolbar.....	73, 75
Skins .....	147	Properties .....	86
Controller.....	154	Structure.....	83
Controls.....	106	Window.....	85
Copy.....	56, 96	Event	
Copying Snapshots.....	172	Initialization Order.....	60
CPU		Loop Prevention .....	134
Load Display.....	74	Loops .....	61, 88, 103
Usage .....	63	Loops Enable.....	134
Create backup file before saving .	61	Processing.....	134
Create Constant.....	123	Signals.....	133
Create Control .....	123	Exit .....	55
Creating Macros.....	112	External Sample Editor .....	65
Creating wires.....	130	External Sync .....	57, 93
Custom-made control .....	159		
Customized fader .....	162	<b>F</b>	
Customized panels .....	162	Factory Content path .....	64
Cut.....	55, 95	Fader.....	139, 156
		File	
<b>D</b>		missing.....	179
Debug .....	58	Files.....	78
Delete.....	56, 96	Filter Envelope.....	48
Deleting wires.....	130		

First Steps .....	23
FM Overdrive.....	29
Frame Style.....	119
Function page.....	87, 99
Function page; Macro .....	115
Functions List Box .....	191

## G

Gate .....	156
Global Auto Save .....	18
Globally disable event loops	61, 134
Global Pitch .....	88
Graph	
BG .....	106
Fill .....	106
Line.....	106
Grid.....	106, 203

## H

Header .....	73
Hierarchy of Reaktor.....	81
Hints .....	131
Hold Ctrl.....	109

## I

IC	
Receive .....	128
Send.....	128
Identical Samples .....	178
Ignore Tempo Change .....	54, 58
Import MIDI File .....	54
Incremental.....	153, 162
Indicator .....	106
Info Page .....	90, 104, 117
Initialization Order.....	60
In Port.....	128
Insert snapshot.....	172

Instrument .....	94
Header.....	73, 96
Object.....	94
Properties .....	98
Internal Sample Rate.....	89
Iteration module .....	133

## K

Key	
Control.....	160
Tracking .....	48
Knob .....	156, 157

## L

Label.....	87
Instrument .....	99
Legacy Mode .....	89
Level .....	88
LFO.....	25
Linking Snapshots.....	170
List of Open Windows.....	72
Load Data into Table .....	204
Local Identifier .....	21
Local IP Address.....	21
Local Port .....	21
Lock/Unlock Panel .....	43, 97
Lock Voices.....	100
Loop	
Editor .....	192
MIDI File.....	58
Lower Limit .....	127
Lower Note Limit .....	109

## M

Macro .....	111
Collection .....	208
Properties .....	115

Main toolbar .....	73	Morphing .....	176
Managing Snapshots .....	170	Mouse Functions.....	50
Map		Moving Objects.....	50
Keyboard View .....	190	Mrph Ctrl .....	109
List View .....	189	MRPH Time .....	177
Master Level.....	88	Multi-Sampling .....	181
Master Tune .....	88	Mute .....	95, 114, 124
Max.....	127, 140	All .....	87
Automation ID .....	155	audio during Core compilation..	62
Unison V .....	101	Port.....	123
Maximum processor usage .....	63		
Measure CPU Usage.....	58	<b>N</b>	
Menu.....	53	Networked Computers .....	19
MIDI		New Ensemble.....	53
Activity lamps .....	97	Note Shift .....	109
Clock Out .....	58	Num Animations .....	
Control.....	160	106, 107, 118, 119, 165	
Data Types .....	160	Number of Undos.....	62
File.....	54		
In Device.....	92, 93, 109, 110	<b>O</b>	
In lamp.....	74	Open .....	53, 73
Learn.....	57, 75, 161	Open Sound Control .....	19
Note .....	154	Options.....	61
Out.....	162	Options List Box .....	191
Out Device .....	110	Order	
Out lamp.....	74	of Audio Processing.....	136
Panel .....	152	of Event Processing.....	134
Sources .....	126	OSC .....	19
Min .....	127, 140	Client.....	22
Unison V .....	101	Ensemble .....	93
Minimize .....	71, 96	Identification .....	21
Missing Samples.....	179	Member List .....	22
Module .....	121	Monitor .....	22
Adding.....	121	Receive .....	128
Context Menu .....	124	Send.....	128
Ports .....	122	Settings .....	60, 110
Properties .....	125	Source .....	154
Sorting.....	59	Synchronization.....	21
Mono.....	124		
Morph, snapshot.....	92		



System Setup .....	20
Target .....	154
Out Port.....	128
Overwrite snapshot.....	172

## P

Padecho .....	26
Panel.....	138
Color .....	91
Controls .....	106, 118, 139
Editing.....	138
Operation .....	156
to MIDI .....	153
Paste.....	56
PgUp/PgDn .....	160
Picture	
Borders.....	107, 119
Index .....	106, 107, 118
Properties .....	163
Pitch Former .....	180
Playerbox .....	65
Play MIDI File.....	54, 58
Plug-in.....	14
Size Functions .....	18
Poly Aftertouch .....	154
Ports .....	95, 113, 122
Pre-listening.....	80
Preferences .....	60
Preview.....	164
Primary Macros.....	111
Primary Structures .....	120
Properties .....	96

## R

Randomize .....	175
Randomizing Snapshots.....	175
Range of Values .....	126
REAKTOR 4 Legacy Mode .....	89

Reaktor as Plug-in.....	14
Reassign .....	102
Recall	
by MIDI.....	88, 103, 169
by Parent .....	102
Recalling Snapshots .....	169
Receive.....	128
Channel .....	92, 93, 109, 110
MIDI .....	152
Recent Ensembles .....	55
Recorderbox .....	67
Recorder Settings .....	67
Redo .....	55
Reload last ensemble at start-up....	
61, 62	
Remap to single key .....	191
Remote to MIDI .....	153
Renaming Snapshots.....	172
Reset All Tool Window Positions..	70
Resizeability .....	164, 167
Restore ... with Ensemble .....	61
Resynth .....	180
Right mouse button.....	51
Root Key .....	183
Run/Stop Audio .....	58, 74

## S

Sample Rate.....	56, 74, 89
Samples .....	178
Analysis .....	180
Editor .....	181
Loop Player .....	32
Management.....	178
Map Editor .....	184
Maps .....	181
Sampling and Resynthesis.....	178
Save .....	73
Ensemble .....	53
Ensemble As .....	53
Instrument As.....	96, 114

Saving .....	43	Spin Control .....	158
Maps .....	184	Start/Restart Clock.....	75
Screen-sets .....	70, 71	Step .....	140
Select A.....	177	Stepsize.....	127
Select All .....	56	Store by Parent.....	102
Selectionng Objects .....	50	Store Map with Ensemble .	179, 195
Select Picture.....	163	Structure .....	120
Send .....	128	Icon.....	106
Set Protected .....	57	Icon, Macro .....	118
Settings Menu .....	56	Toolbar.....	73, 76
Set transpose to zero.....	191	Sustain Control .....	109
Set Unprotected .....	57	Switch .....	143, 145, 157
Show		Switches .....	128
Hints .....	76	Sync Clock .....	93
Module Sorting .....	59	SynthOne .....	24
Value .....	141	System Menu .....	58
Show/Hide			
Browser .....	70	<b>T</b>	
Hints .....	65	Table	
Panel.....	70	Control Mode.....	204
Playerbox .....	65	Draw Mode.....	204
Properties .....	69	Files path.....	65
Recorderbox .....	67	Modules.....	196
Sample Map Editor.....	70	Select Mode .....	204
Snapshots .....	70	Targa .....	165
Toolbox .....	65	TCP/IP.....	19
Signal Processing.....	132	Tempo .....	75
Skin .....	147	Terminals .....	95, 113, 128
Skins, button.....	150	Text box.....	158
Skins, fader.....	147	TGA.....	165
Snap Isolate .....	174	Tile	
Snapshot .....	97, 168	Horizontally .....	71
ID.....	169	Vertically .....	71
Master for Plug-in .....	88	Toggle .....	157
Morphing.....	176	Toolbar .....	73
Soft Takeover.....	152, 162	toring Samples with Modules....	179
Solo .....	95	Total Recall .....	15
Sound installations.....	19	Transparency .....	164, 165
Source Modules.....	125	Trigger .....	156
Sources .....	125		

Tune.....	88
Instrument .....	100
Tutorial Ensembles.....	24

## **U**

Undo .....	55, 62, 73
Unison.....	97
Mode.....	101
Spread.....	100
Upper Limit.....	127
Note .....	109
User Content path.....	64

## **V**

Value Grid .....	203
View .....	65
Visible .....	107, 119
Voice & MIDI	
Slave To .....	100
Voices.....	97, 100
Allocation.....	100
Assign .....	102
Combiner .....	95
Steal mode.....	102

## **W**

Wire .....	35
Wires.....	129
Wires, creating.....	50
workspace.....	81

## **X**

XY Field .....	158
----------------	-----