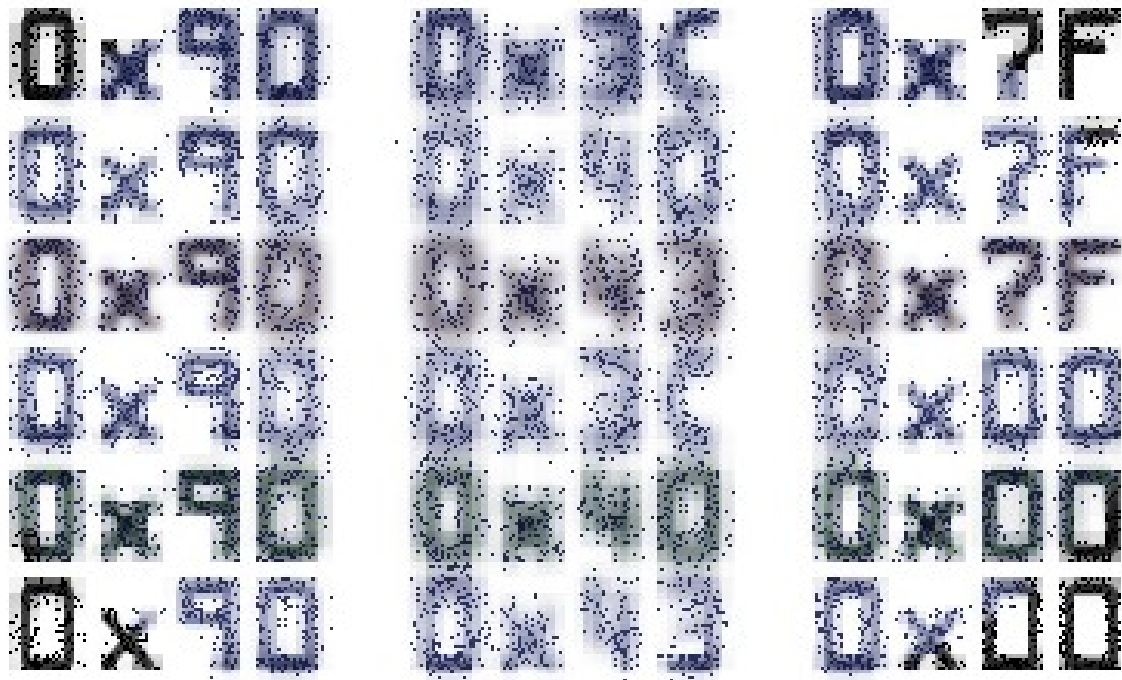


piz midi plugins



By Insert Piz Here->

Introduction

The Piz MIDI VST Plugins are a collection of VST 2.4 plugins for Windows, Mac OS X, and Linux.

They are updated frequently (sort of), so check the website to make sure you have the latest version:

<http://thepiz.org/pizmidi>

The plugins do not have their own graphical user interface at this time, so the GUI will be provided by the VST host.

For Windows, DarkStar has provided GUI versions based on mGUI. Get them here:

<http://asseca.com/wiki/MGUI/MdaPizmidi>

If you like these plugins, you can give me money. There is a PayPal donation link at the website.

General Information

To install, put the plugin files in your VST folder. Most hosts let you define where this is. For the plugins to work, a VST host is needed, and the host must be able to handle MIDI output from VST plugins.

Supported:		As VST Instruments Only:	Not Supported:
Aodix	MaizeStudio	SONAR	Chainer
AudioMulch	Max/MSP		Logic
Bidule	MiniHost		Orion
Buzz/Buzé/etc (via Polac VST loader)	MU.LAB		Podium
Cantabile	Pure Data		Pro Tools
Console	REAPER		Project 5
Cubase/Nuendo	SAVIHost	As VST Effects Only:	Renoise
energyXT	SynthEdit	Cubase LE 4 (no instrument rack)	Samplitude
FL Studio	Temper	You need to force them to load as effects via pizmidi.ini.	Studio One
forte	Tracktion		
Jost	Usine		Untested:
Kore	VFX/V-Machine		ACID
Live	VSTHost		Phrazor
			Psyce
			Tunafish
			Vegas
			n-Track Studio
			everything else.

Most plugins will load as "VST effects" except in hosts that need them as "VST instruments." You can override this in pizmidi.ini (see below).

Default bank

Every plugin can load a set of default settings on startup. To set this up, your VST host must be able to save banks in *.fxb format. Simply configure the plugin with your desired settings, and then save a file (*plugin name*).fxb in the "pizmidi" folder:

- **Windows:** C:\Documents and Settings*(username)*\pizmidi\ (or in the same folder as the plugins)
- **Linux:** ~/.pizmidi/
- **Mac OS X:** /Users/*(username)*/Library/pizmidi/

pizmidi.ini

The settings file, **pizmidi.ini**, is a plain text file saved in the pizmidi data folder as described above. Settings are per-host. For each host, on one line put **host=(hostname)**, where "(hostname)" is the name of the host as reported through the VST interface (`getHostProductString()`).

Then under each **host=** line, put any of the following options (one per line):

force inst	plugins will load as instruments.
force effect	plugins will load as effects.
audio outs	plugins will have a stereo audio output.
no audio outs	plugins will not have any audio outputs.
audio ins	plugins will have a stereo audio input.
no audio ins	plugins will not have any audio inputs.
bottom octave=n	base octave number for plugins which display note names (default is -2).

Included plugins

[midi16CCRouter](#) - Changes up to 16 CCs to 16 other CCs.

[midi2qwerty16](#) - Use MIDI to trigger fake keystrokes to the focused window.

[midiAlias](#) - Note values are reflected between a "Nyquist Note" and a "Zero Note," similar to the effect of audio sample aliasing.

[midiAudioToCC](#) - Converts audio amplitude to CC messages.

[midiBlackKeyFilter](#) - Separately adjust velocity of black and white keys, and/or send them on different channels.

[midiCCModulator](#) - Use one CC to modulate another CC (or note velocity) in various ways: add/subtract/multiply/etc, plus logic gates (AND, NAND, XNOR, etc).

[midiCCReset](#) - Remembers CC values and sends them out again at selected times.

[midiCCStepper](#) - Quantize CC events by value or in time.

[midiCCToggle](#) - For changing momentary CC buttons into toggles.

[midiChannelFilter](#) - Simple single-channel filter.

[midiChannelize](#) - Transforms everything to the selected channel.

[midiChordSplit](#) - For splitting polyphonic material to multiple midi channels based on selected note ranges for each voice.

[midiChs_noGUI](#) - Channel filter/changer like energyXT1 "MIDI chs" component.

[midiConverter3](#) - Change and convert between CC, Channel Pressure, Poly Aftertouch, Program Change, Pitch Bend, NRPN/RPN, Note On/Off, and other messages in various ways.

[midiDelay](#) - Delay notes and/or CCs with feedback.

[midiDuplicateBlocker](#) - Remembers all MIDI message values it gets on each channel, and blocks any repeated values until a different value is received.

[midiExactDelay](#) - Simple delay (no dry signal, no feedback) with flexible and precise delay time.

[midiFingered](#) - Selectively sends standard portamento on/off CC messages to emulate "fingered portamento" for synths that don't have it.

[midiForceToKey](#) - Enforces a scale, and transposes within the scale (optionally based on MIDI notes on a selected channel).

[midiForceToRange](#) - Transposes notes by octave into the selected range.

[midiInvertNotes](#) - "Mirrors" notes centered around the selected note.

[midiKeySplit](#) - Simple keyboard splitter.

[midiKeySplit4](#) - Four-zone keyboard splitter.

[midiNotchFilter](#) - Isolate, remove, or channelize specific MIDI messages.

[midiNoteGroups](#) - Assign play and/or choke groups to MIDI notes.

[midiNotesToCC](#) - Converts note number and velocity to CC messages.

[midiNoteToggle](#) - Notes (optionally within a selected range and/or channel) are toggled. Play a note once to turn it on, play it again to turn it off.

[midiNRPNConverter](#) - Converts between NRPN/RPN and CC messages, with 16 conversion slots.

[midiOverlappingNoteKiller](#) - Prevents overlapping notes at the same pitch.

[midiPitchBendQuantize](#) - Quantizes pitch bend messages to up to 25 values.

[midiPitchBendToNotes](#) - Changes pitch bend plus note messages into notes at the "bent" pitch (may be useful for using Suzuki QChord with non-multitimbral instruments).

[midiPolyphony](#) - For spreading polyphony across MIDI channels. Polyphony per channel can be set from 1 to 16 notes for more flexibility. You can also use it to convert polyphonic input to monophonic output.

[midiProbability](#) - Add probability-based effects to incoming MIDI events, optionally synced to host timeline.

[midiProgramChange](#) - Send Program Change and/or Bank Select messages.

[midiScaleChanger](#) - Transposes notes by individual scale degrees and/or globally, to change one scale into another. Among other things, this allows you to play any scale using only the white keys.

[midiSimpleLFO](#) - One CC LFO with some "simple" options.

[midiSostenuto](#) - Like a piano sostenuto pedal, but for MIDI notes instead of piano dampers.

[midiStrum](#) - Sequentially delays notes of an input chord to simulate strumming.

[midiStuckNoteKiller](#) - Sends note-off messages to end hanging notes when the host stops (for hosts that don't do this themselves).

[midiSwing](#) - Delays and/or offsets velocity of and applies probability to notes (and optionally also CCs) at every other step of a selected note value.

[midiTranspose](#) - Transposes notes on a selected channel, with selectable range up to ± 72 semitones.

[midiTriggerList](#) - Use one note or CC to trigger up to 8 other notes and/or CCs.

[midiVelocityScale](#) - Scale/reverse/offset velocity of input notes (Note On and/or Note Off).

midi16CCRouter

Changes up to 16 CCs to 16 other CCs. Optionally responds to Program Change. Notes, unselected CCs, and other MIDI messages are sent through unaffected.

Parameters:

- **CC In ##** - Input CC number for this slot, between 0 and 127
- **CC Out ##** - Output CC number for this slot, either "No Change" (same as input), or between 0 and 127
- **Ch. In** - Input channel to listen on, either "Any" or 1-16
- **Ch. Out** - Output channel, either "No Change" or 1-16
- **PC Channel** - Channel to listen to for Program Change messages, either "Any", 1-16, or "None"

midi2qwerty16

Use MIDI to trigger fake keystrokes to the focused window. Any used MIDI messages will also be blocked from the plugin's output.

Parameters:

Per slot (16 of each)

- **##: data** - MIDI message to use as the trigger for this slot.
Options:
 - **Off** - Slot will be ignored
 - **Note** - Selected Note-On message
 - **ProgCh +** - Program Change message one program higher than the previous
 - **ProgCh -** - Program Change message one program lower than the previous
 - **CC (127)** - Selected CC with value 127
 - **CC (0)** - Selected CC with value 0
 - **CC (>64)** - Selected CC with any value over 64
 - **CC (<64)** - Selected CC with any value under 64
 - **CC slider > 100** - Selected CC with any value over 100, and won't be triggered again until after the value drops below 100
 - **CC slider < 27** - Selected CC with any value under 27, and won't be triggered again until after the value goes over 27
- **note/cc** - Note or CC number, between 0 and 127 (if data is "Note", note name will also be shown). Not used for ProgCh settings.
- **modifiers** - Key modifiers to apply to the sent keystroke, either "None" or any combination of Shift, Ctrl, Alt, and Windows keys.
- **key** - Key to send.

Global

- **channel** - MIDI channel to listen on, "Any" or 1-16.
- **program** - Selects a program (for automating program changes).

midiAlias

Note values are reflected between a "Nyquist Note" and a "Zero Note," similar to the effect of audio sample aliasing.

Parameters:

- **Nyquist Note** - The highest allowed note value. Input notes higher than this will be reflected downward.
- **Zero Note** - The lowest allowed note value. Input notes lower than this will be reflected upward.
- **Pre Shift** - Number of semitones to shift input notes before applying the reflection.
- **Power** - Turns the plugin on or off.

midiAudioToCC

Converts audio amplitude (RMS) to CC messages. Audio is sent through the plugin unaffected.

Parameters:

- **Input Gain** - Gain applied to the input signal before conversion to MIDI. This is only used for internal calculations and does not affect the output signal.
- **CC Out L** - For Mono mode, this is the output CC; for Stereo mode, this is the output CC for the left channel.
- **CC Out R** - For Stereo mode, this is the output CC for the right channel; not used for Mono mode
- **Channel Out** - Output MIDI channel
- **Mode** - Mono (summed left and right) or Stereo operation
- **Inertia** - Wimpy smoothing of output CC messages
- **Rate** - Controls the rate of output CC messages, from about 300 to 1000 per second

midiBlackKeyFilter

Separately adjust velocity of black and white keys, and/or send them on different channels.

Parameters:

- **InputCh** – Channel to affect: Any, or 1-16
- **WhiteCh** – Channel to send white keys on: No Change, or 1-16
- **WhiteVel** – Velocity adjustment in percent for white keys (0-200%)
- **BlackCh** - Channel to send black keys on: No Change, or 1-16
- **BlackVel** - Velocity adjustment in percent for black keys (0-200%)

midiCCModulator

Use one CC to modulate another CC (or note velocity) in various ways.

Parameters:

- **Mode**

Options:

- Off - The plugin won't do anything
- CC -> CC - Modulate a CC
- CC -> Velocity - Modulate the velocity of input MIDI notes

- **Input** - The input CC to modulate, either "Any CC"/"Any Note" or 0-127
- **Modulator** - The CC used to modulate the input, or "Manual" which uses a VST parameter instead of a CC
- **Manual** - When Modulator is set to "Manual", this parameter is used as the modulator
- **Output** - Output CC, either "Same as Input" or 0-127, ignored for "CC -> Velocity" mode
- **Amt/Thresh** - Modulation amount, threshold, or nothing, depending on the ModType setting
- **ModType** - The algorithm to apply to the Input and Modulator.

Options:

- Invert - Simply inverts the input values, modulator is not used
- Add - Scales Modulator by Amt and adds it to Input, limited to 127 maximum output
- Average - Averages Input and Modulator, Amt crossfades between Input and the average
- WrapAdd - Same as Add, but values over 127 wrap around instead
- Subtract - Scales Modulator by Amt and subtracts it from Input, limited to 0 minimum output
- Add/Subtract - Like Add or Subtract, but treats Modulator as a bipolar value between -64 and +63
- Multiply - Normalizes Modulator to a maximum value of 2.0, scaled by Amt, and multiplies Input by it
- Compare - Outputs the higher of Input or Modulator
- Limit - Outputs the lower of Input or Modulator
- Bite - When Input is greater than Modulator, subtracts Modulator (scaled by Amt) from Input
- Mangle - When Input is less than Modulator, does a bipolar addition like Add/Subtract mode
- Gate - When Modulator is below Thresh, output is 0
- Block - When Modulator is at or above Thresh, output is 0
- AND - When both Modulator and Input are at or above Thresh, output is 127, otherwise 0
- NAND - When both Modulator and Input are at or above Thresh, output is 0, otherwise 127
- OR - When either Modulator or Input are at or above Thresh, output is 127, otherwise 0
- NOR - When either Modulator or Input are at or above Thresh, output is 0, otherwise 127
- XOR - When either Modulator or Input (but not both) are at or above Thresh, output is 127, otherwise 0
- XNOR - When either Modulator or Input (but not both) are at or above Thresh, output is 0, otherwise 127
- Match - When Modulator is equal to Input, output is 127, otherwise 0
- bitwise OR - Input | Modulator
- bitwise AND - Input & Modulator
- bitwise XOR - Input ^ Modulator
- << - Shifts the bits of Input left by Modulator amount
- >> - Shifts the bits of Input right by Modulator amount
- Randomize - Applies a random offset to Input between 0 and Modulator amount, scaled by Amt

- **OutLevel** - Output level from 0-200%
- **Channel** - MIDI Channel to listen on
- **Thru** - What to do with MIDI messages other than the Input and Modulator
 - Off - All incoming messages are blocked; only the output message is sent
 - Block ModCC - All incoming messages are sent through, except the selected Modulator CC which is blocked
 - All Thru - All incoming messages are sent through

midiCCReset

Remembers CC values and sends them out again at selected times.

Programs:

Each VST program saves its own set of CCs (128 CCs x 16 channels), and these can optionally be sent when selecting a program. All VST parameters (except ManualTrig and Clear) are common to all programs.

Parameters:

- **ManualTrig** - Move this parameter to maximum to manually send the saved CCs.
- **RecordTrig** - When set to "On" the saved CC values will be sent when the host starts recording.
- **ProgramTrig** - When set to "On" the saved CC values will be sent when changing programs on the plugin (either manually or with Program Change messages, if PC Listen is enabled).
- **Capture** - When set to "On" any incoming CC events will be saved as the default values.
- **Clear** - Move this parameter to maximum to clear any saved CC values.
- **Channel** - MIDI channel to listen/send on. When set to "All" the plugin will store and send every CC on every channel (128x16 = 2048 values).
- **PC Listen** - When "On" the plugin will respond to Program Change messages on the above selected channel.
- **Duplicates** - Determines whether duplicate CC messages will be sent on preset change.
 - **Allow** - Always send the whole preset
 - **Block** - Only send the preset values that are different from the last output values
- **Thru** - When "On" all incoming MIDI is passed through; otherwise it is blocked, and the triggered CCs are the only output from the plugin.

midiCCStepper

Quantize CC events by value or in time.

Parameters:

- **Val Quantize** - On/Off switch for value quantization
- **ValQ Steps** - Number of steps for value quantization. From 2 (CC values rounded to either 0 or 127) to 128 (no change)
- **Time Quantize** - On/Off switch for time quantization
- **Note Value** - Tempo-based value for time quantization, from 128 (128th note) to 1 (whole note)
- **Host Sync** - For time quantization. When On, the note value quantization steps will be aligned to the host timeline. Otherwise the timing will be based only on input events.

- **Beat Offset** - Offset the quantization grid from the host timeline, from 0-100% of the Note Value. Only applies if Host Sync is On.
- **SendCCAfter** - When enabled, CCs received after the time step are sent at the next time step.
- **Low CC** - Lowest CC to affect; every CC from this one to High CC will be affected
- **High CC** - Highest CC to affect
- **CC Shift** - Changes the CC number of all affected CCs by this amount. For example, incoming CC 1 with CC Shift of 8 will be sent out as CC 9. Shift can be -127 to +127; numbers will wrap around.
- **Channel** - MIDI Channel to listen on. Other channels are sent through unaffected.
- **NotesThru** - Optionally block notes from being sent through the plugin.

midiCCToggle

For changing momentary CC buttons into toggles. When a value of 64 or higher is received, the outgoing value will toggle between 0 or 127. Incoming values below 64 are blocked.

Parameters:

- **Power** - On/Off switch
- **Low CC** - Lowest CC to affect; every CC from this one to High CC will be affected
- **High CC** - Highest CC to affect
- **Channel** - MIDI Channel to listen on. Other channels are sent through unaffected.

midiChannelFilter

Only passes through data on the selected channel.

Parameters:

- **Channel** - Channel to filter. When set to "All" the plugin will have no effect (all channels will be passed).

midiChannelize

Transforms everything to the selected channel.

Parameters:

- **Channel** - Channel to transform everything to. When set to "No Change" the plugin will have no effect.

midiChordSplit

For splitting polyphonic material to multiple midi channels based on selected note ranges for each voice. This is different from a straight keyboard splitter because the ranges can overlap, and the plugin attempts to choose the most appropriate channel.

Global Parameters:

- **Power** - On/Off switch for the whole plugin.
- **InChannel** - Input channel.
- **NotePriority** - How to decide which notes to end when polyphony limit is exceeded:
 - **Steal Oldest** - The note with the oldest start time will be ended.
 - **Steal Last** - The note with the most recent start time will be ended.
 - **Steal Quietest** - The note with the lowest original velocity will be ended.
 - **High** - Higher notes have priority, lowest pitched note will be ended.
 - **Low** - Lower notes have priority, highest pitched note will be ended.
 - **Centered** - Notes closest to the "center" (median) of held note have priority.
 - **Random** - Note to steal is chosen at random.

Per-channel settings, with N from 1 to 16:

- **ChNPower** - On/Off switch for channel N.
- **ChNLow** - Lowest allowed note for channel N. Can't be set higher than ChNHigh.
- **ChNHigh** - Highest allowed note for channel N. Can't be set lower than ChNLow.
- **ChNPoly** - Maximum number of notes allowed on channel N (1-64 notes).

midiChs_noGUI

Channel filter/changer like energyXT1 "MIDI chs" component.

Parameters:

- **1** -> through **16** -> - Transform each incoming MIDI channel to the selected output channel. Set to "--" to block the channel.
- **Reset** - Setting this parameter to maximum sets all output channels to the same as the input channels.
- **Clear** - Setting this parameter to maximum sets all output channels "--" (blocked).

midiConverter3

Change and convert between CC, Channel Pressure, Poly Aftertouch, Program Change, Pitch Bend, NRPN/RPN, Note On/Off, and other messages in various ways. Pitch Bend and (N)RPN conversions will use the full 14-bit values when possible.

Parameters:

Input section

Here you select an input message to convert.

- **Input Type** - Incoming message to convert. See also "In Param 1" and "In Param 2" which further define the message. For note input types, note-offs and 0-velocity note-ons are treated interchangeably. For LSB modes, the low 7 bits of the 14-bit message are used as the input message.

Options:

- **CC**
 - **14-bit CC**
 - **Program Change**
 - **Channel Pressure**
 - **Poly Aftertouch**
 - **Pitch Bend**
 - **Pitch Bend (LSB)**
 - **NRPN**
 - **NRPN (LSB)**
 - **RPN**
 - **RPN (LSB)**
 - **Note On #**
 - **Note On Velocity**
 - **Note Off #**
 - **Note Off Velocity**
- **In Param 1** - Selects input CC number, note number, or (N)RPN MSB.
 - **In Param 2** - Selects input (N)RPN LSB.
 - **Channel In** - Selects input channel.

Mapping section

These parameters determine how the input values are mapped to output values.

- **Low Input** - Lowest allowed input value (except for Stretch/Stretch mode, see below). Low Input cannot be higher than High Input. If you try to set it that way, High Input will be automatically set equal to Low Input.
- **High Input** - Highest allowed input value (except for Stretch/Stretch mode, see below). Low Input cannot be higher than High Input. If you try to set it that way, Low Input will be automatically set equal to High Input.
- **Map Mode**
 - Options:
 - **Clip/Limit** - Values outside Low & High Input are discarded; values inside are then limited to Low & High Output.
 - **Clip/Stretch** - Values outside Low & High Input are discarded; values inside are then stretched or compressed to fit between Low & High Output.

- **Stretch/Stretch** - The full input range 0-127 is compressed to fit between Low & High Input and then is stretched or compressed again to fit between Low & High Output.
- **Low Output** - Lowest possible output value. When Low Output is higher than High Output, the values are reversed, and this will be the highest output value.
- **High Output** - Highest possible output value. When Low Output is higher than High Output, the values are reversed, and this will be the lowest output value.
- **Offset** - This value is added to the output value after the above mapping.

Output section

Here you set what the input message is converted to. For 14-bit to 7-bit conversions, rounding is used. For 7-bit to 14-bit conversions, the input value is used for both bytes, except when the value is below 64, in which case the LSB is set to 0.

- **Output Type**

Options:

- **Discard** - No conversion, just block the input message.
- **CC** - Mapped value will be used as the value of a CC message set by Out Param 1, optionally (set by Out Param 2) with smoothing.
- **14-bit CC** - Mapped value will be used as the value of a 14-bit CC message pair set by Out Param 1, optionally (set by Out Param 2) with smoothing.
- **Program Change**
- **Channel Pressure**
- **Poly Aftertouch** - Mapped value will be used as the value of a polyphonic aftertouch message with note number set by Out 1.
- **Pitch Bend** - Mapped value will be used as the value of a pitch bend message, optionally (set by Out Param 1) with smoothing.
- **Pitch Bend (LSB)** - Mapped value will be used as the LSB of a pitch bend message with MSB set by Out Param 1.
- **NRPN**
- **NRPN (LSB)**
- **RPN**
- **RPN (LSB)**
- **Note On #** - Mapped value will be used as the note number for a note-on message with velocity set by Out Param 1. Optionally (set by Out Param 2) also immediately send a corresponding note-off.
- **Note On Velocity** - Mapped value will be used as the velocity for a note-on message with note number set by Out Param 1.
- **Note Off #** - Mapped value will be used as the note number for a note-off message with velocity set by Out Param 1.
- **Note Off Velocity** - Mapped value will be used as the velocity for a note-off message with note number set by Out Param 1.
- **Clock** - Input message will trigger a MIDI Clock tick (0xF8).
- **Out Param 1** - Selects output CC number, note number, or (N)RPN MSB.
- **Out Param 2** - Selects output (N)RPN LSB, or Pitch Bend / CC smoothing amount.
- **Channel Out** - Selects output channel.
- **Thru** - What to do with other MIDI messages.

Options:

- **Block Converted** - Unselected messages are sent through; selected input messages are blocked.
- **Thru All** - All messages are sent through, except for conversions where the output is the same message as the input.
- **Converted Only** - Unselected messages as well as selected input messages are blocked; only the output messages are sent.

midiDelay

Delay notes and/or CCs with feedback.

Parameters:

- **TempoSync** - When On, DelayTime will be measured in musical note values, following host tempo; when Off, DelayTime will be absolute time in milliseconds.
- **DelayTime** - Either tempo-synced values from 64th note to 3 bars (TempoSync On), or 0-2000ms (TempoSync Off).
- **Feedback** - "Amount" of delayed signal to be delayed again. At 0%, the input will only be delayed once; at 100% the input signal will repeat indefinitely.
- **FdbkLimit** - Optionally limit the number of times each event will be sent through the feedback loop. Either "Off" (unlimited) or 1-16.
- **Wet** - Level of the delayed events, from 0-100% of the internal value. This is applied only at the output, after the feedback path.
- **Dry** - Level of the input events (notes and/or CCs) passed through to the output.
- **Mode** - Notes Only, or Notes & CCs. Be careful when delaying CC events with feedback.
- **Channel** - MIDI channel on which to delay events; either "All" or 1-16.

midiDuplicateBlocker

Remembers all MIDI message values it gets on each channel, and blocks any repeated values until a different value is received.

Parameters:

- **Mode** - Choose the type of messages to affect, or turn off the plugin.
Options:
 - **Off**
 - **CCs**
 - **Notes** - Disregards velocity
 - **PitchBend**
 - **ProgChange**
 - **ChanPressure**
 - **PolyAftertouch**
 - **All - All of the above**
- **Channel** - MIDI channel to affect. When set to "All" the plugin will block duplicates on every channel.

midiExactDelay

Simple delay (no dry signal, no feedback) with flexible and precise delay time, measured in seconds, tempo-synced note values, or samples.

Parameters:

- **TimeMode** - Choose the units of measurement (and corresponding parameters) to use for delay time:
 - **Seconds** - Absolute delay time, adjustable from 0.001 to 100.999 seconds using the Seconds/Tenths/Hundredths/Thousandths parameters.
 - **NoteValue** - Tempo-based delay time, set by the Bars/Beats/NoteValue/Ticks parameters.
 - **Samples** - Absolute delay time in samples, set by the the 1000/100/10/samples parameters.
- **Seconds** - Adds 0-100 whole seconds to the delay time.
- **Tenths** - Adds 0.0-0.9 seconds to the delay time.
- **Hundredths** - Adds 0.0-0.09 seconds to the delay time.
- **Thousandths** - Adds 0.0-0.009 seconds to the delay time.
- **Bars** - Adds 0-32 bars (based on time signature) to the delay time.
- **Beats** - Adds 0-32 beats (based on time signature) to the delay time.
- **NoteValue** - Adds a musical note duration from a 128th note to a dotted half (set to "--" for 0 delay).
- **Ticks** - Adds 0-100 ticks (960 ticks per quarter note) to the delay time.
- **1000 samples** - Adds 0-10000 samples in increments of 1000 to the delay time.
- **100 samples** - Adds 0-900 samples in increments of 100 to the delay time.
- **10 samples** - Adds 0-90 samples in increments of 10 to the delay time.
- **samples** - Adds 0-9 samples to the delay time.
- **Output** - Scale the values of output messages by this amount (0-100%).
- **Mode** - Type of events to delay: Notes Only, Notes & CCs, or Everything.
- **Channel** - MIDI channel on which to delay events; either "All" or 1-16.

midiFingered

Selectively sends standard portamento on/off CC messages to emulate "fingered portamento" for synths that don't have it. For this to work, the target instrument must have a portamento switch that responds to CC 65.

Parameters:

- **Power** - On/Off switch
- **Channel** - MIDI channel, either "All" or 1-16.

midiForceToKey

Enforces a scale, and transposes within the scale (optionally based on midi notes on a selected channel).

Parameters:

- **C / C# / D / D# / E / F / F# / G / G# / A / A# / B** - These parameters define the scale as on/off switches for each note. When an "off" note is received, it will be changed to a nearby "on" note, based on the setting of the Mode parameter.
- **Mode**
 - Options:
 - **Off** - Turns off the plugin; input notes set to "off" will be output unchanged.
 - **Nearest** - "off" notes will be changed to the nearest "on" note, whether higher or lower, with priority given to lower notes.
 - **Always Up** - Transposes "off" notes up to the next "on" note.
 - **Always Down** - Transposes "off" notes down to the next "on" note.
 - **Block** - Blocks "off" notes completely.
- **Transpose** - Transposition by scale degree; each note will be moved this number of steps (from -50 to +50) within the scale, after being forced to the scale.
- **Tr.Channel** - When not set to "None", notes on this channel will set the Transpose parameter based on their distance from the selected Root Note.
- **Root Note** - Note to use for scale transposition, when using the Tr.Channel.
- **N.Channel** - When not set to "None", notes on this channel activate the note on/off switches.
- **N.Ch.Mode** - Options for the effect of input notes on N.Channel:
 - **Momentary** - Switches are "on" only while input notes are held down (note-on is "on," note-off is "off").
 - **Toggle** - Input note-ons toggle the switches between "on" and "off."
- **Program** - This parameter simply selects a VST program.

midiForceToRange

Transposes notes by octave into the selected range. If the range is incomplete (less than 12 notes), notes not included in the range will be ignored.

Parameters:

- **LowNote** - Lowest note of the target range. Notes below this note will be transposed up.
- **HighNote** - Highest note of the target range. Notes above this note will be transposed down.
- **Channel** - Only notes on this channel will be affected; either "Any" or 1-16.

midiInvertNotes

"Mirrors" notes centered around the selected note.

Parameters:

- **Center Note** - Notes on either side of this one will be transposed to the other side of this note; the center note itself does not get transposed.
- **Power** - Turns the plugin on or off.

midiKeySplit

Simple keyboard splitter. Splits the keyboard into two zones, each of which can be sent on a different channel and/or transposed separately.

Parameters:

- **1: OutChannel** - Output channel for the first zone (below the Split Note).
- **1: Transpose** - Number of semitones to transpose the first zone, from -48 to +48.
- **Split Note** - First note of the second zone. Set this parameter to 0 for MIDI learn.
- **2: OutChannel** - Output channel for the second zone (Split Note and above).
- **2: Transpose** - Number of semitones to transpose the first zone, from -48 to +48.
- **InChannel** - Only notes on this channel will be affected.

midiKeySplit4

Splits the keyboard into four zones, each of which can be sent on a different channel and/or transposed separately.

Parameters:

- **[1, 2, 3, 4]: OutChannel** - Output channel for the numbered zone.
- **[1, 2, 3, 4]: Transpose** - Number of semitones to transpose the numbered zone, from -48 to +48.
- **[1/2, 2/3, 3/4]: Split Note** - First note of the second zone in the numbered pair. Set this parameter to 0 for MIDI learn.
- **InChannel** - Only notes on this channel will be affected.

midiNotchFilter

Isolate, remove, or channelize up to 4 specific midi messages.

Parameters:

- **Message ##** - Type of message for this slot:
 - **None** - Slot is disabled.
 - **CC** - Control Change message; optionally select channel and CC number below.
 - **Note** - Note on and off messages; optionally select channel and note number below.
 - **Pitch Bend** - Pitch Bend message; optionally select channel below.
 - **Chan.Pressure** - Channel Pressure (Aftertouch) message; optionally select channel below.

- **Aftertouch** - Polyphonic Aftertouch message; optionally select channel and note number below.
- **Prog.Change** - Program Change message; optionally select channel below.
- **##.Channel** - MIDI channel for this slot.
- **##.Value** - CC or note number for this slot (for CC, Note, and Aftertouch messages only).
- **Mode**
 - Options:
 - **Block (Channelize)** - When Out Channel is not set, the selected messages are blocked; otherwise they will be sent on the selected channel instead of the input channel.
 - **Pass** - Only the selected messages will be sent through, and will be transformed to the Out Channel if set; all other messages are blocked.
- **Out Channel** - Sets the output channel for affected messages.

midiNoteGroups

Assign play and/or choke groups to MIDI notes.

Parameters:

Global

- **Use ProgCh** - When on, the plugin will respond to Program Change on the selected channel.
- **Channel** - Input and output channel to use.
- **Thru** - When on, other MIDI messages will pass through, otherwise only the grouped notes will be sent.

Per slot (8 of each)

- **Note ##** - Choices: Select a MIDI note (0-127)
 - Set to "**Off**" to disable the slot, or
 - Set to "**Learn Next**" to use the next received MIDI note as the selected note
- **##: ChokeGroup** - When receiving any note assigned to this group, any other playing notes in the group will end. If any notes are in the same play and choke groups, they will be ended first and then immediately retriggered.
- **##: PlayGroup** - When receiving any note assigned to this group, all other notes in the group will also play.

midiNotesToCC

Converts note number and velocity to CC messages.

Parameters:

- **Note to CC** - Selects the destination CC number for note number conversion. The pitch of incoming notes will be used as the value of this CC. Set to "Off" to disable.
- **High Note** - The highest input note that will be used. This range acts as a filter for both note number and velocity conversion.
- **Low Note** - The lowest input note that will be used.
- **High NCC Value** - Highest possible output value for the note number CC. The incoming range defined by High Note & Low Note is stretched/compressed to this range.
- **Low NCC Value** - Lowest possible output value for the note number CC.
- **NCC Rest Value** - If enabled (not set to "Off") this value will be sent when a note-off is received. Only applies when Note to CC is also set.
- **Velocity to CC** - Selects the destination CC number for velocity conversion. The velocity of incoming notes will be used as the value of this CC.
- **High Velocity** - The highest input velocity that will be used (1-127). This range acts as a filter for both note number and velocity conversion.
- **Low Velocity** - The lowest input velocity that will be used (1-127).
- **High VCC Value** - Highest possible output value for the velocity CC. The incoming range defined by High Velocity & Low Velocity is stretched/compressed to this range.
- **Low VCC Value** - Lowest possible output value for the velocity CC.
- **VCC Rest Value** - If enabled (not set to "Off") this value will be sent when a note-off is received. Only applies when Velocity to CC is also set.
- **In Channel** - "Any" or 1-16.
- **Out Channel** - Output CCs are sent on this channel, either "Same as input" or 1-16.
- **Thru**
Options:
 - **Off** - Only the output CCs will be sent.
 - **Unconverted** - Unused MIDI messages will be sent through; only notes that have been converted are blocked.
 - **All** - All MIDI messages will be sent through, including notes used for conversion.

midiNoteToggle

Notes (optionally within a selected range and/or channel) are toggled... play a note once to turn it on, play it again to turn it off.

Parameters:

- **Power** - Turn note toggling on or off.
- **Low Note** - Lowest note to toggle.
- **High Note** - Highest note to toggle.
- **Channel** - Channel on which to toggle notes.

midiNRPNConverter

Converts between NRPN/RPN and CC messages, with 16 conversion slots. Slots are processed top to bottom. Multiple slots can be set to the same input message to convert one message to multiple messages.

Parameters:

Global

- **Input** - Input message type (CC, NRPN, NRPN(LSB), RPN, or RPN(LSB)). For the LSB modes, the MSB is ignored.
- **Output** - Output message type (CC, NRPN, NRPN(LSB), RPN, or RPN(LSB)).
- **Channel In** - Input channel on which to look for messages. Either "Any" or 1-16.
- **Channel Out** - Channel on which to send output messages. Either "No Change" (same as input) or 1-16.

Per slot (16 slots)

- **##:Coarse In** - Select the input CC number (0-127) or MSB of the input (N)RPN in hex (0x0000 - 0x3f80). When set to "None" the slot is disabled.
- **##:(N)RPN In** - Sets the LSB of the input (N)RPN; first set the Coarse In, then the full (N)RPN is displayed here in hex. Not used when Input is set to CC. Set this to 0 to enable NRPN Learn.
- **##:Coarse Out** - Select the output CC number (0-127), or MSB of the output (N)RPN. When set to "None" the input message is simply blocked.
- **##:(N)RPN Out** - Sets the LSB of the output (N)RPN. Not used when Input is set to CC.

Another global parameter

- **Thru** - What to do with other MIDI messages.
Options:
 - **Block Converted** - Unselected messages are sent through; selected input messages are blocked.
 - **Block All** - Unselected messages as well as selected input messages are blocked; only the output messages are sent.
 - **Thru All** - All messages are sent through, except for conversions where the output is the same message as the input.

midiOverlappingNoteKiller

Prevents overlapping notes at the same pitch.

Parameters:

- **Power** - On or Off.

midiPitchBendQuantize

Quantizes pitch bend messages to up to 25 values.

Parameters:

- **q+12 through q-12** - Quantization steps. 25 values of pitch bend can be set, and other values will snap to the nearest step.
- **Channel** - MIDI channel to affect (Any, or 1 – 16).

midiPitchBendToNotes

Changes pitch bend plus note messages into notes at the "bent" pitch (useful for playing Suzuki QChord with non-multitimbral instruments). A pitch bend range of 12 is assumed, but will also be set from the standard RPN message for pitch bend range.

Parameters:

- **Power** - On or Off.
- **Channel** - Output MIDI channel. When set to "No Change" the notes (and any other MIDI messages) are sent on the original channel; otherwise they are all combined to the selected channel.

midiPolyphony

For spreading polyphony across midi channels, i.e. if you play 4 notes, each one will go to a different channel. Polyphony per channel can be set from 1 to 64 notes for more flexibility. You can also use it to convert polyphonic input to monophonic output.

Parameters:

- **Power** - On/Off switch for the effect.
- **LowChannel** - First channel to route notes to.
- **HighChannel** - Last channel to route notes to.
- **PolyPerChan** - Maximum number of notes to be sent on each channel (1-64 notes).
- **NotePriority** - How to decide which notes to end when polyphony limit is exceeded:
 - **Steal Oldest** - The note with the oldest start time will be ended.
 - **Steal Last** - The note with the most recent start time will be ended.
 - **Steal Quietest** - The note with the lowest original velocity will be ended.
 - **High** - Higher notes have priority, lowest pitched note will be ended.
 - **Low** - Lower notes have priority, highest pitched note will be ended.
 - **Centered** - Notes closest to the "center" (median) of held note values have priority.
 - **Random** - Note to steal is chosen at random.
- **Retrigger** - When on, held notes that have been stolen (based on NotePriority) will be retriggered again when polyphony becomes available.
- **InChannel** - Input channel
- **SustainPedal** - What to do with sustain pedal messages (CC 64) on the selected InChannel:
 - **Pass Thru** - CC 64 will be sent to all channels between LowChannel and HighChannel
 - **Use** - CC 64 will be used to sustain notes internally, and will not be sent through.

- **ChanMode** - Options for determining how the channel for each note is chosen:
 - **Reuse** - Lowest channel is always used until polyphony exceeds the PolyPerChan setting (old method).
 - **Prefer New** - Consecutive notes will cycle through the channels.

midiProbability

Add probability-based effects to incoming midi events, optionally synced to host timeline. Has two slots which can each be set to affect a different step on the host timeline. Slots are processed top-to-bottom, with the first slot having priority.

Parameters:

- **Power** - Selects what types of MIDI messages to affect:
 - **Off**
 - **On (Notes)**
 - **On (Notes & CCs)**
 - **On (Everything)**

Per-slot parameters (2 slots)

- **##:Step** - Musical note value from 3 bars to 64th note. Probability effects will only apply to events near (Required "nearness" is determined by the TimeFocus parameter) this step, starting at 0 on the host's timeline. Set Step past 64 to "All Notes" to ignore this value and simply affect every event. See also StepShift below.
- **##:StepShift** - Shifts the above step from 0% (no shift) to 100% (full step length). For example, at StepShift=50% and Step=8, this slot will affect events that occur halfway between every 8th note.
- **##:Mode** - Selects the type of effect for this slot. Whether or not the effect actually applies to each specific event is determined by the Prob parameter (see below). Modes:
 - **Off** - Slot is disabled.
 - **Skip** - Events will be skipped.
 - **OctaveUp** - Notes will be transposed up one octave. Has no effect on other events.
 - **OctaveDown** - Notes will be transposed down one octave. Has no effect on other events.
 - **RandTransp** - Notes will be transposed by a random amount between -12 and +12 semitones. For other events, the first data byte will be shifted by this amount.
 - **Transpose** - Notes will be transposed by the amount set by the Transpose parameter below.
 - **RandVeloc** - Velocity of notes will be shifted by a random amount between -64 and +64. For other events, the second data byte will be shifted by this amount.
 - **OffsetVel** - Velocity of notes will be shifted by the amount set by the VelOffset parameter below.
 - **Channelize** - Events will be sent on the channel set by the ChnlizeChan parameter below.
 - **Multi** - Events will be processed by all "Multi" parameters (see below).
- **##:Prob** - Probability for events to be affected by this slot, from 0% (never) to 100% (always).

Global parameters

- **TimeFocus** - Affects how "wide" each step is, or how close an event must be to a step before it is affected. Higher values mean the step is more "focused" and events must be closer to the step.

- Multi probabilities: These only apply to slots that have Mode set to Multi. Any number of Multi effects can simultaneously apply to any given event.
 - **Multi-Skip** - Probability for events to be skipped.
 - **Multi-OctUp** - Probability for notes to be transposed up an octave.
 - **Multi-OctDn** - Probability for notes to be transposed down an octave.
 - **Multi-RndTrn** - Probability for notes to be transposed by a random interval.
 - **Multi-Transp** - Probability for notes to be transposed by the amount set by Transpose.
 - **Multi-RndVel** - Probability for notes to be shifted in velocity by a random amount.
 - **Multi-OffsetVel** - Probability for notes to be shifted in velocity by the amount set by VelOffset.
 - **Multi-Chnlze** - Probability for notes to be sent on ChnlizeChan.
- **Transpose** - Interval up to +/-36 semitones to transpose affected notes by.
- **VelOffset** - Amount up to +/-64 to shift the velocity of affected notes by.
- **ChnlizeChan** - Channel to use for channelized events, either "Random" or 1-16.
- **InputChan** - Only events on this channel will be affected; can be "All" to affect all channels.

midiProgramChange

Send Program Change and/or Bank Select messages.

Internal Programs:

The plugin's internal VST programs can be used to store 128 Program/Bank combinations, which will be sent when switching to that internal program. All parameters other than Program, Bank MSB, and Bank LSB are shared between all internal programs.

Parameters:

- **Program** - Selects the program number to send with Program Change messages, or set to "Off" to disable sending PC. When Mode is set to Direct, changing this parameter will itself send the PC messages.
- **Bank MSB** - Selects the MSB of the bank number to send with Bank Select messages, or set to "Off" to disable sending Bank MSB. When Mode is set to Direct, changing this parameter will itself send the Bank MSB messages.
- **Bank LSB** - Selects the LSB of the bank number to send with Bank Select messages, or set to "Off" to disable sending Bank LSB. When Mode is set to Direct, changing this parameter will itself send the Bank LSB messages.
- **Mode** - Options:
 - **Direct** - in addition to using triggers, also send messages continuously when the above parameters are adjusted;
 - **Triggered** - only send messages when triggered by the parameters below.
- **PC Trig.** - When parameter goes to maximum, send a Program Change message.
- **PC Incr.** - When parameter goes to maximum, increase Program by one and send a Program Change message.
- **PC Decr.** - When parameter goes to maximum, decrease Program by one and send a Program Change message.
- **Bank Trig.** - When parameter goes to maximum, send Bank Select MSB and/or LSB.

- **PC Listen** - When on, the plugin will respond to incoming Program Change messages, which will select an internal program and send the values for that program.
- **Channel** - Input/output channel.
- **Thru** - Whether or not to send through other MIDI messages.

midiScaleChanger

Transposes notes by individual scale degrees and/or globally, to change one scale into another. Among other things, this allows you to play any scale using only the white keys.

Example: Change C Major to F# Minor

- Disable C#, D#, F#, G#, A#
- Set E, A, and B to -1
- Set Root Note to -6 (F#)

Input notes in C Major and they come out as F# Minor.

Parameters:

- **C / C# / D / D# / E / F / F# / G / G# / A / A# / B** - Transpose each input note by up to +/-12 semitones, or set to "--" to disable a note. These settings apply to every octave.
- **Root Note** - Transposes the whole scale by up to +/-48 semitones.
- **OctaveWrap** - If the total offset (Root Note + scale degree transposition) would place a note outside of the MIDI range (0-127), it will either be transposed by octave back into the range (OctaveWrap On) or blocked (OctaveWrap Off).
- **OutChannel** - Non-disabled notes will be output on this channel.
- **AltChannel** - When not set to "None", disabled notes will be sent on this channel.
- **Program** - This parameter simply selects a VST program.

midiSimpleLFO

One CC LFO with some "simple" options.

Parameters:

- **Waveform** -
 - **Sawtooth** - Descending linear ramp
 - **Ramp** - Ascending linear ramp
 - **Triangle** - Linear ramp up/down
 - **Square** - Evenly-spaced high & low values only
 - **Sine** - Sinusoidal
 - **Slow Random** - Sends a random value once every cycle
 - **Fast Random** - Sends a lot of random values every cycle
- **CC Out** - CC number for the LFO output. Incoming events of this CC are blocked.
- **Channel Out** - Channel for the LFO output.
- **Freq Mode**
 - Options:
 - **Tempo** - Frequency in musical times, based on host tempo.

- **Hz** - Frequency in Hz, independent of host tempo.
- **Frequency** - In Tempo mode, 32 bars to 32nd note; In Hz mode, 0-27 Hz. At 0 Hz a single CC value will be sent only when triggered by a note (see Triggering modes).
- **Phase** - Oscillator phase from -180 to +180 degrees. Can be modulated in real time.
- **Amplitude** - Sets the LFO's output range, from 0 to 127. The range is centered at 63.
- **Offset** - A constant added to all the values, from -64 to +63, for adjusting the "center line" of the waveform.
- **Triggering**
 - Modes:
 - **Notes (Mono)** - The LFO will start when a note is received, and will continue as long as any note is playing.
 - **Notes (Poly)** - The LFO will restart every time a note is received, and will continue as long as any note is playing.
 - **Beat Sync** - The LFO will be active only when the host is playing, and will be in sync with the host timeline.
 - **Free Running** - The LFO will be always active, independent of note input and host timeline.
- **Power** - On/Off switch for the LFO.

midiSostenuto

Like a piano sostenuto pedal, but for midi notes instead of piano dampers. Sostenuto is a selective sustain pedal; only notes that are held down at the moment the pedal is depressed will be sustained.

Parameters:

- **Pedal CC** - MIDI CC to use for the sostenuto pedal. This should be 66 according to the MIDI specification, but you can use any CC here.
- **Low Note** - Lowest note that can be affected by sostenuto.
- **High Note** - Highest note that can be affected by sostenuto.
- **Pedal Position** - Instead of using a CC, you can use this parameter as the pedal.
- **Channel** - Only notes on this channel will be affected.

midiStrum

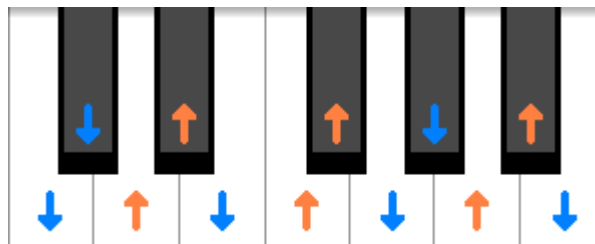
Sequentially delays notes of an input chord to simulate strumming.

Parameters:

- **Mode**
 - Options:
 - **Complicated** - Chord notes and strum patterns (downstroke or upstroke) are sent separately
 - **Simple** - Any notes that arrive at the same time will be strummed with a downstroke
- **NoteLength** - Sets whether the length of strummed chords is based on the chord notes (InChannel) or the strum notes (StrumChannel).
- **Speed** - Speed of the strum. At 0% the total strum time will be Max Time; at 100% speed the strum will be instantaneous.

- **Max Time** - Maximum total strum time, either in milliseconds (100-3000 ms) or in musical tempo-based values (64th note to 4 bars) depending on the setting of TempoSync.
- **TempoSync** - Sets whether strum time is absolute or related to tempo.
- **VelToSpeed** - Sets how much effect strum note velocity will have on strum speed (higher velocity --> faster strum).
- **VelToVel** - Sets how much effect strum note velocity will have on the velocity of notes in the strummed chord.
- **DownVel** - Sets the base velocity for downstroke chords.
- **UpVel** - Sets the base velocity for upstroke chords.
- **VelRamp** - Applies a change in velocity over the course of the strum.
- **Randomness** - Applies randomness to the velocity and duration of notes within the strummed chord.
- **Accel** - Speed up or slow down over the course of the strum. Overall strum time remains the same.
- **In Chan** - Channel to use for chord note input (which will be automatically strummed in Simple mode).
- **Strum Chan** - Channel to use for strum stroke input. Can be the same as In Chan, but make sure you don't want to play any notes in the Strum Octave.
- **Strum Octave** - Set the location of the notes which are used to trigger strums. Downstrokes and upstrokes are mapped within this octave as follows:

C, C#	down
D, D#	up
E	down
F & F#	up
G & G#	down
A & A#	up
B	down



- **Out Chan** - Output channel for strummed chords.
- **Audition** - When set to "Always," notes on the In Chan will be sent through unstrummed. This is useful when building chords for Complicated mode, but should be turned off when strumming. When set to "While Strumming," a low-velocity note is sent through only if strum is sounding, for a "hammer-on" kind of effect.
- **Use Sustain** - When on, sustain pedal (CC 64) on In Chan will be used to hold chord notes, and those CC 64 events will not be passed through to the MIDI output.

midiStuckNoteKiller

Sends note-off messages to end hanging notes when the host stops (for hosts that don't do this themselves).

Parameters:

- **Power** - On/Off switch.

midiSwing

Delays and/or offsets velocity of and applies probability to notes (and optionally also CCs) at every other step of a selected note value.

Parameters:

- **Mode** - Off, Notes Only, or Notes & CCs.
- **NoteValue** - Swing will apply to events near every other step of this value, from 3 bars to a 64th note. "Near" is defined by TimeFocus.
- **Swing** - From 0% up to MaxSwing.
- **TimeFocus** - Sets how much to focus on the selected time step, by narrowing the width of the affected range of time.
- **Time Shape** - Sets the shape of the "swing envelope" around the selected time step.
 - **Parabolic** - Events in the middle of the range will be delayed the most, with gradually less delay approaching the edges.
 - **Flat** - All events inside the range will be delayed by the same amount.
 - **Linear Up** - The amount of delay increases linearly between the beginning and end of the range.
 - **Linear Down** - The amount of delay decreases linearly between the beginning and end of the range.
- **TimeOffset** - Shifts the position of the time step by up to one full step in either direction.
- **Velocity** - Apply a velocity offset to swung notes, from -126 to +127.
- **Probability** - This is the probability that any given swung note will be played. When less than 100%, some notes will be skipped.
- **Randomness** - Applies an amount of randomness to delay times and velocity.
- **MaxSwing** - Determines the maximum possible delay for swung notes:
 - **100%** - 2:1 ratio (for tied-triplet swing), delay is 1/3 of NoteValue.
 - **150%** - 3:1 ratio (for e.g. dotted 8th/16th note swing), delay is 1/2 of NoteValue.
 - **200%** - 5:1 ratio, delay is 2/3 of NoteValue.
 - **250%** - 11:1 ratio, delay is 5/6 of NoteValue.
 - **300%** - ∞ :1 ratio, delay is equal to NoteValue
- **NoteOff** - What to do with note off messages:
 - **Equal Delay** - Whenever possible, delay the note-off by the same time as the note-on was delayed, so that the note length stays the same
 - **Stingy Delay** - If possible, don't delay the note-off. If and only if the delay time is longer than the original note, the note-off will be delayed by the same amount as the note-on.
 - **Legato** - Swing will apply to note-offs based on their position, so that the space between consecutive notes remains about the same.
- **Channel** - Input channel to affect, either "All" or 1-16.

midiTriggerList

Use one note or CC to trigger up to 8 other notes and/or Ccs.

Parameters:

Global

- **Trigger Type**

Choices:

- **Learn Next** - will wait for the next note or CC to determine the trigger settings
- **Note On/Off**
- **CC 127/0** – 127 is on, 0 is off
- **CC > 0** – Any value greater than 0 is on, 0 is off

- **Trigger Num** - Note or CC number to be used as the trigger. To use MIDI learn, set Trigger Type to "Learn Next" (Trigger Num will display "Waiting...") and send the desired message.
- **Send CC Off** - When on, the trigger "off" message (Note-Off or CC value of 0) will trigger a value of 0 for any slots set to "CC" type.
- **Use ProgCh** - When on, the plugin will respond to Program Change on the selected channel.
- **Channel** - Input and output channel to use.

- **Thru**

Choices:

- **All Thru**
- **Block Trigger**
- **Block All** - only MIDI generated by the plugin will be sent

Per slot (8 of each)

- **Slot ## Type**

Choices:

- **Learn Next** - will wait for the next note or CC to determine the slot's settings
- **Off**
- **Note**
- **CC**

- **Slot ## Num** - Note or CC number for the slot.
- **Slot ## Value** – Velocity/CC value for the slot. "As Input" means the velocity of the trigger note will be used.

midiTranspose

Transposes notes on a selected channel, with selectable range up to ± 72 semitones, and option to transpose immediately (splitting sounding notes).

Parameters:

- **Transpose** - Actual transpose amount in semitones, controllable by Pitch Bend and/or Transp CC, if enabled below.
- **Range** - Range of the Transpose parameter, from 0 to ± 72 semitones (± 6 octaves).
- **Immediate** - When enabled, on any change of the transpose amount, currently playing notes will be ended and restarted at the new pitch. Otherwise, they will continue at the old pitch until a note off event.
- **InChannel** - Input channel(s) to affect, All or any single channel.
- **PitchBend**
 - **Thru** - Pitch Bend events on InChannel are just passed through the plugin
 - **Use** - Pitch Bend events on InChannel are used to control the Transpose parameter, and are blocked from the output.
- **Transp CC** - CC number on InChannel to use to control the Transpose parameter.

midiVelocityScale

Scale/reverse/offset velocity of input notes (Note On and/or Note Off).

Parameters:

- **Velocity 1 / Velocity 2** - The input range 1 - 127 will be mapped to the range Velocity 1 - Velocity 2. If Velocity 1 is greater than Velocity 2, the range is reversed.
- **Offset** - This value is added to the outgoing velocity after mapping to the range defined above.
- **ReleaseVel1 / ReleaseVel2** - Same as Velocity 1/2 but for note-off velocity.
- **Rel.Offset** - This value is added to the outgoing note-off velocity after mapping to the range defined above.