Xhip

User Manual

Version 7, released 12 April 2012.

TABLE OF CONTENTS

Introduction

Xhip lore

Xhip was originally designed as a basic synthesizer used inside a tracker ("acid tracker") around 1999. Before that it had existed in pieces as small tools for generating samples to be used in more common trackers.

In 2003 it existed as a stand-alone application ("acid synth") when someone suggested that it should be converted to a VSTi. It needed a new name as the synthesizer was originally designed for simple chip-sounds with a little extra and not so much for TB-303 bass-lines or whatever else might be associated with *acid*.

Xhip was born

In 2004 someone suggested it should be released publicly on a web page with news about it posted on KVR. At the time it had no GUI and much more limited capabilities.

How do you say Xhip? Well, the same way you'd say xello, or xlevator, or xurbo xharger of course. Personally I use a sound like rough sand-paper scratching along a plank. Incidentally I wonder if Xhip can produce this sound ... ?

That's enough of a history lesson for now; Let's move on to the...

Xweet xubtractive xudio xynthesis!

With the addition of a second oscillator, super-saw oscillator unison, two filters, a wave shaper, pairs of modulators (envelopes and LFOs); Features like ring-mod, x-mod, filter input FM, saturation and global unison Xhip has become far more than its original intention.

Xhip produces the highest quality dual-oscillator polyphonic synthesizer sounds similar to classic poly synthesizers of the late 80s. Strings, synth pads, basses, leads, organs, pianos, percussion, bells, vocals, sound effects and more are demonstrated in the included bank.

Graphical User Interface

Display

The display shows a text readout of the currently edited parameter as it is adjusted on the GUI.

It will also display messages associated with errors or actions taken via the buttons, menus or switches.

Voice count

The number of active voices are displayed here numerically.

LEDs

The display includes a few LEDs which are part-way between generally useful and space-filler.

Peak meter

A row of LEDs originating at the center of the display below the text contains a peak meter. Each LED in the peak meter represents a 3 decibel step in peak amplitude. The red LED illuminates erroneously between -3 and 0 decibels as opposed to above 0 decibels.

MIDI receive

This LED will illuminate briefly upon any MIDI input to the plug-in.

MIDI sustain

This LED will illuminate while MIDI hold/sustain (CC #64) is held.

Voice activity

These sixteen LEDs will illuminate individually while each corresponding voice (1 to 16) is active providing a rough impression of the way voices have been allocated by the synthesizer.

Buttons

Various buttons are provided to activate internal (not managed by the host) features of the plug-in.

Load, save (preset, bank)

These buttons will pop up a dialog box allowing a Xhip format preset or bank to be selected and loaded or saved.

Clear

This button will over-write the currently selected preset with the internal “init” state. The internal init preset is a pure unfiltered square wave without modulation.

Load (wav)

This button will pop up a dialog allowing a Microsoft RIFF WAVE format file to be loaded, if it is supported by the plug-in. Please see the section on PCM functionality for further detail.

Map CC

This button acts as a typical “MIDI learn” button and will begin scanning for any subsequent parameter adjustment. Any subsequent MIDI CC input will immediately map to that parameter.

Unfortunately there is no option to save or load a predefined mapping nor is there an option to display or clear the current map.

Resources

Skins

Unfortunately version 7 of the plug-in does not support any skinning features.

Bitmaps

Although skinning features are not supported the bitmap resources used by the plug-in are stored in the system application data directory. This is generally located at “C:/ProgramData/xhip”. The bitmaps are stored as 32-bit-ARGB TGA with RLE compression. If desired these bitmaps can be edited to suit user preference. “Paint dot net” http://www.getpaint.net/ is a freeware application for Windows with support for this file format.

Specification

Synthesizer

This table outlines the sections/components of the synthesizer and their parameters. All synthesizer parameters are fully adjustable via VST parameter automation.

Oscillators	Glide (time, mode [pitch, Hz] [logarithmic, linear]) Range (32, 16, 8, 4, 2) Sync (A => B, B => A, gate) X-mod mode (A! => B, B! => A, A! <=> B!) Pitch modulation (envelope [A, B], modulator [A, B])
A & B	Waveform (pulse, ramp, triangle/sine, S&H noise, PCM) Width X-mod depth
A	Width modulation (envelope [A, B], modulator [A, B])
B	Keyboard tracking Tuning (coarse, fine) Pitch modulation (envelope [A, B], modulator [A, B])
Mixer	Oscillator A (level, invert) Oscillator B (level, invert) Ring-mod enable Noise level
Filter	Mode ([12 dB, 24 dB] [low, band, high, notch]) Trigger (hard, soft) Frequency Q Keyboard tracking F-mod Q-mod Saturation Frequency modulation (envelope [A, B], modulator [A, B])
Waveshaper	Mode (logarithmic, exponential, absolute) Depth Symmetry Tone
Post-shaper Filter	Mode ([6 dB, 12 dB] [low, band, high, notch]) Frequency Q Keyboard tracking

Output	Volume Volume modulation (modulator [A, B]) Volume envelope (selectable [A, B, gate]) Panning Panning modulation (modulator [A, B])
Envelopes	Attack, decay, sustain, release Constant attack time enable Keyboard tracking Trigger (gate, modulator [A, B], reset, delay [A, B])
Modulators	Shape (pulse, ramp, sine, noise, random) Rate Delay Width Keyboard tracking Bias (negative, bipolar, positive) Range (low, audio, tempo-synced) Key-sync enable

Control

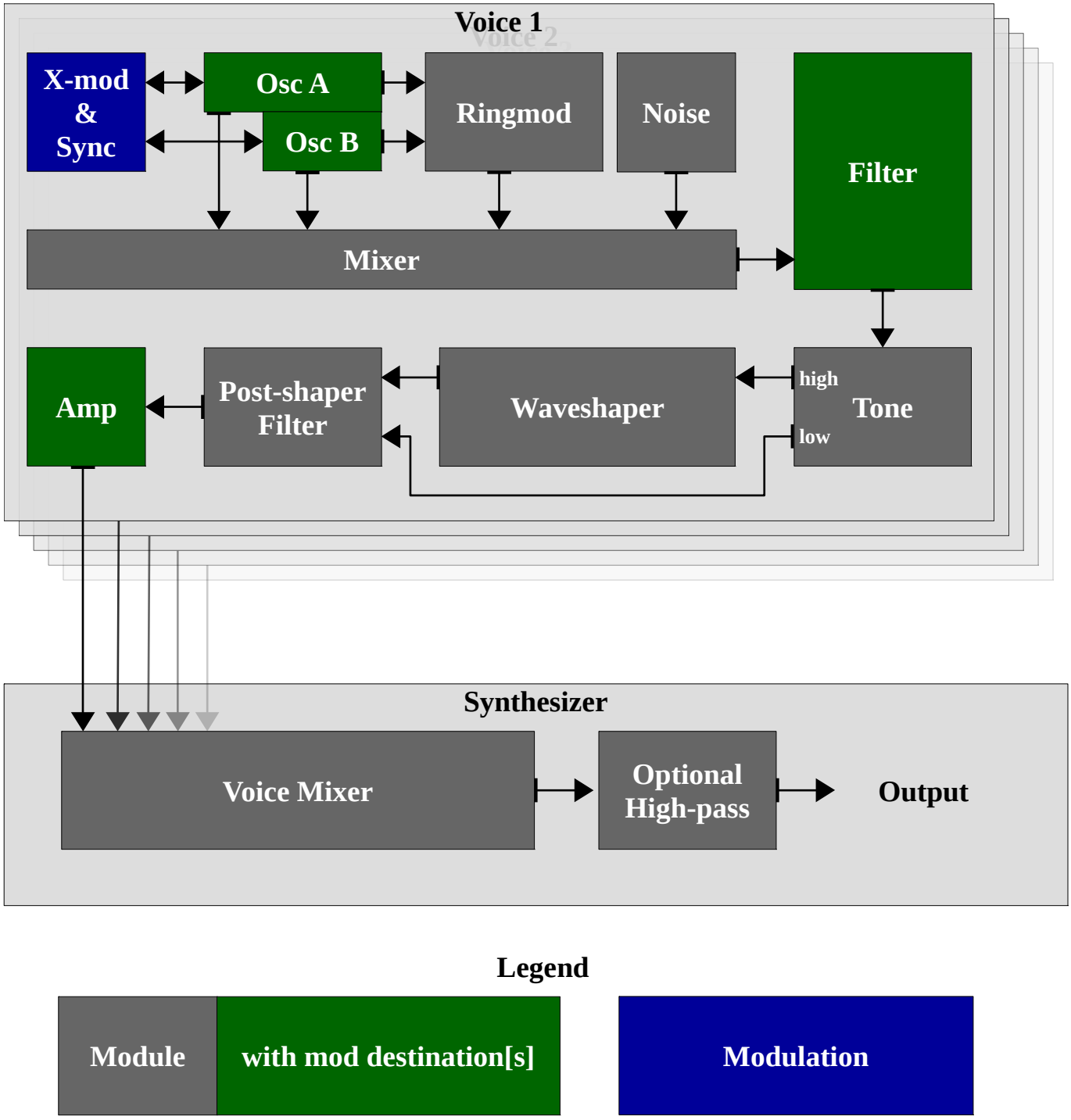
The control section is not a part of the synthesizer itself. Parameters in the control section are not available via the VST interface and can not be adjusted via VST parameter automation.

Control	Tune (coarse, fine) Volume Bender range Glide mode Voices Velocity route (destination, scale, offset) Bender route (destination, scale, offset) Unison (voices, detune, randomization, distribution) Flags / mode switches

In addition the control section contains a knob used to adjust globally (for all instances) the number of unison waveforms used for the *super-saw* ramp oscillator unison. This parameter defaults to 16 and is not saved in presets or projects. It should not normally be used.

Synthesizer

Signal Flowchart



Oscillators

The oscillators section contains parameters affecting both oscillators. Options for sync and X-mod routing as well as pitch modulation including range and glide are available here.

Glide

The glide mode parameter includes both logarithmic and linear modes applied to either the voice pitch or frequency (Hz) as well as an option to disable the glide thereby removing its processing cost.

In the logarithmic mode a low-pass filter is applied to the pitch or frequency signal. This is the most common glide effect present in synthesizers. In this mode the pitch will change abruptly at first and then more slowly proportionate to the distance from the destination pitch. Due to this property the time to glide between any two semi-tones on the keyboard is equal.

In the linear mode an integrator is applied. This glide effect is seen more often in digital synthesizers and is less common than a logarithmic type. The rate of pitch change is constant in this mode. Due to this property the time to glide between two semi-tones depends upon their distance from one another.

The timing between both logarithmic and linear modes is equal for a glide between semi-tones exactly one octave apart.

The linear mode when applied to frequency (Hz) applies an automatic adjustment on note-on to ensure the time between any two semi-tones is equal. This is required in order to deal with the fact that a natural consequence of the linear effect applied to frequency is that the rate is proportionate to the position on the keyboard as if key-tracking were applied. The distance between two semi-tones in Hz at a lower octave (C4 to D4) is half as much as it is in a higher octave (C5 to D5.)

Range

Range or *foot pitch* is a term inherited from pipe organs which represents the transpose in octaves. 8' pitch represents a neutral setting while each halving or doubling represents a transpose upward or downward by one octave.

Sync (mode)

The oscillators may be configured with sync in either direction. Additionally a **gate** mode is available which will sync both oscillators upon note-on.

X-mod (mode)

X-mod or *cross modulation* takes the output of one or both of the oscillators and uses it to frequency modulate the oscillators.

The X-mod is applied to pitch (exponential) which means the oscillator's tuning will be shifted as depth is increased. It is best used to apply mild audio-rate modulation where the modulator should be anti-aliased. When used in combination with sync the synced oscillator can be forced to stay in tune with the source despite the application of a high level of modulation.

An interesting effect can be achieved by applying sync **A => B** and X-mod **A! => B**.

This means "**A!**" ("feedback A") and "**=> B**" ("modulate B") with the result. With this setting the X-mod parameter for oscillator A acts as feedback while the parameter for oscillator B acts as modulation depth.

Pitch modulation

A mod destination is provided for the pitch shared between the oscillators which may be modulated by envelopes or modulators to create pitch shifting effects such as vibrato.

Some parameters are present for both oscillators A and B.

Waveform

Off	The oscillator may be disabled by selecting off.
Pulse	Standard pulse waveform.
Ramp	Up-ramp often referred to as a *saw* which may optionally be run in unison.
Triangle	Standard triangle may optionally be wave-shaped to approximate a sine.
Staircase	White noise sampled at four times the oscillator frequency.

Width

The width control has a varied use dependant upon which waveform is selected for the oscillator.

Pulse	Used to adjust the duty-cycle of the pulse waveform. At 50% the output will be equal and otherwise known as *square* and can be adjusted to a narrow pulse at either side of the parameter's range. The width modulation input from envelopes and modulators will take effect only when used with the pulse waveform.
Ramp	Used to enable and adjust the detune amount for the ramp waveform *super-saw* oscillator unison. With width centered the unison will be disabled and a single ramp will play. As the parameter is adjusted in either direction unison will take effect and detuning will increase.
Triangle	Used to blend between triangle and triangle wave-shaped to sine. The sine waveshaper is a very cheap approximation (asymmetric parabolic) and will carry a significant amount of harmonic content although at a lower level than the triangle waveform. THD 3.3%.

The width control does not take effect for the staircase or PCM waveforms.

X-Mod (depth)

Controls the amount of X-mod applied to the oscillator. The modulation source signal is selected in the O**scillators** section with the **X-mod (mode)** parameter.

Oscillator A

Width modulation	Modulation destination for PWM.

Oscillator B

Keyboard tracking	Variable keyboard tracking independent from oscillator A.
Coarse & fine tuning	Tuning offset.
Pitch modulation	Modulation destination for tuning offset.

Mixer

Oscillator A & B (level, invert)

These parameters control the level of each oscillator. Phase-inversion is also possible using the invert parameter.

Ring-mod (enable)

This switch replaces the usual addition of the oscillators (A + B) with multiplication (A × B). While enabled the mix level for oscillator B acts as a depth with 100% at 0 dB.

Noise (level)

This parameter controls the level of uniform white noise mixed with the other waveforms.

Filter

The filter is a single or dual (series) Kerwin-Huelsman-Newcomb filter. This filter type is often referred to as a *state-variable filter* and provides low-pass, band-pass and high-pass outputs.

Both the low-pass and high-pass modes have a slope of 12 dB or 24 dB.

The band-pass mode has a slope of 6 dB or 12 dB. This is one half of the filter slope applied on the low side of the band with one half on the high side.

In place of slope the width or steepness of the notch depends upon the **Q** parameter.

Both single and dual (series) configurations are provided. In the series configuration the second stage **Q** is equal to slightly less than $\frac{1}{\sqrt{2}}$ **.** This reduces the amplitude of filter resonance generated by the first stage.

Trigger

The filter trigger mode can be used to set the filter's *phase* or in other words to input a *spike* (impulse) or to *reset* the filter when note-on occurs. This can be used to immediately begin oscillation at note-on and is useful when producing drum sounds or otherwise where the filter's oscillation is used as a primary component of the sound rather than acting solely upon the input signal.

The phase of the band-pass is set to one with the low-pass set to zero. Therefore the band-pass mode will output a *pop* when the trigger is enabled. The low-pass mode will integrate (filter) this *pop*. In the high-pass mode the impulse will be blended between the direct (band-pass) and filtered (low-pass) output according to the Q parameter.

F-Mod, Q-Mod

The F-Mod and Q-Mod parameters control the level of modulation for filter frequency and Q by the signal input to the filter. That is in other words the signal output from the mixer stage.

An example of this effect occurs naturally in the MS10 or MS20 synthesizer with the KORG35 filter. Any signal input to the filter also modulates the cutoff and resonance slightly. (Similar to approximately 10% F-Mod and 50% Q-Mod).

Saturation

Applies non-linearity within the filter leading to change in timbre. TODO: this may require a lot of explanation to cover all the bases.

Waveshaper

The waveshaper is a per-voice distortion after the filter and before the second filter. It is limited to a mild range as it does not do any over-sampling or other anti-aliasing.

Mode

Logarithmic applies the sigmoid $\frac{x}{(|x|+d)}(1+d)$ while **exponential** applies the same sigmoid with a negative coefficient. Exponential mode includes a hard clipping stage to prevent the signal level increasing uncontrollably. **Absolute** mode provides continuously variable half-wave through full-wave rectification.

Symmetry

Used to apply an offset to the signal before it passes into the waveshaper. This can lead to the production of even harmonic content with increasing asymmetry.

Tone

The tone control acts as a frequency splitter where the low-frequency content will bypass the distortion.

This allows the waveshaper to be focused on high-frequency content which is useful for adjusting tonal balance or focusing application on filter oscillation or other content without having low-frequency content dominate the sound.

Post-shaper Filter

The post-shaper filter fills in as both a tone control for the waveshaper as well as for the rest of the synthesizer. It does not provide any modulation inputs although it is capable of keyboard tracking.

Mode

Low-pass and high-pass modes are supplied with both 6 dB and 12 dB slopes. Band-pass and notch modes are provided in two variations with 6 dB slope.

Amplifier

The amplifier provides adjustment of output level and panning as well as a mod-destination for each.

Amplitude

The amplitude envelope can be selected from one of envelope **A** , **B** or g**ate** which is an attack-release envelope with fixed times. Amplitude may also be modulated by both modulator A and B.

Panning

Panning may not be modulated by an envelope.

Panning may be modulated by both modulator A and B.

A bug exists in version 7 of the plug-in where the normalized amplitude of the panning law (-3 dB) is not equal to the amplitude at the exact center position. For this reason use of panning modulation may introduce clicks or pops while the panning parameter crosses zero and so it is not recommended that it be used this way.

Envelopes

The envelopes are standard ADSR envelopes much like those found in numerous other synthesizers.

Envelope timing

Timings are displayed based upon the time it takes to travel **99%** of the step size. For example during the release stage of the envelope the peak of the envelope is **1.0** or **0 dB** and the envelope decays toward **0.0** or **-inf dB**. The time displayed is the length of time during which the envelope will have reached within **1%** of the destination. That is **0.01** or **-40 dB**.

This may make the time displayed for envelopes seem unusually high in comparison to other synthesizers which use a measure based upon a smaller segment of the curve.

Attack stage asymptote

Envelopes are often described referring to the attack stage as *linear.* This is often incorrect. In reality the attack stage is generally the same curve as the decay and release stages however the end point of the curve (called its asymptote) is beyond the peak level of the envelope. In other words the curve is *clipped off* at some point and the decay begins.

If the envelope were allowed to continue to move toward the asymptote it would only reach that point after an infinite amount of time. This is disadvantageous as we likely do not have an infinite amount of time to wait for the decay stage! Common asymptotes are **101%** to **200%** of the peak. This makes the curve appear more linear as varied amounts are c*lipped off*.

The envelopes use an asymptote set at **200%** which is slightly more linear than some synthesizers although not entirely linear. The decay stage starts at **50%** between zero and the asymptote.

Constant attack

Constant attack adjusts the attack time such that regardless of the level the envelope starts from, multiple voices triggered at the same time will always enter the decay stage at the same time.

Trigger

The envelope is normally triggered by the gate when a note is played. It is also possible to trigger the envelope by combination of the gate and a modulator or to delay the envelope trigger until the modulator's delay time has passed. The envelope may also be forced to reset to zero before the attack stage begins if desired although this sudden jump will often lead to clicks and pops.

Modulators

Shape

Pulse	Standard pulse waveform adjustable between narrow pulses and square.
Ramp	Adjustable between up-ramp, triangle and down-ramp.
Sine	Like ramp with a parabolic shaper applied.
Noise	Normalized noise with a low-pass filter applied.
Random	White-noise sampled by ramp & hold with ramp time set by the width parameter.

Width

For the modulators the width parameter takes effect on all waveforms except noise. Pulse can be adjusted between narrow pulses and square. Ramp and sine between up-ramp/s-curve, triangle/sine and down-ramp/s-curve. Random can be adjusted between immediate jumps and linear interpolation.

Rate

In **low** mode a typo lead to the minimum rate being **2003.97 seconds** with the maximum rate at **32.96 Hz**. This can be considered a non-critical “bug” and will be fixed in the next version. Recommendation is to use times between **30 seconds** and **30 Hz** while in **low** mode.

In **audio** mode the rate can be adjusted between **-32 semitones** and **+32 semitones** in **25 cent** steps. Tuning is relative to the current note assuming keyboard tracking is set to **100%**.

In **tempo** mode timing can be adjusted between triplet **32 whole** to **64ths**.

Delay

An exponential attack is applied to abruptly fade-in the modulator. The effect is intended to be quite abrupt and is ideal for delaying the start of a vibrato or other modulation. Delay time can be adjusted between **zero** and **5.5 seconds**.

Bias

This parameter can be used to scale the modulator waveform by **1/2** allowing it to operate from **-1** to **0 negative** or **0** to **+1 positive** rather than the usual **-1** to **+1 bipolar** range.

Control

The control section is not a part of the synthesizer itself. Parameters in the control section are not available via the VST interface and can not be adjusted via VST parameter automation. The parameters in the control section are not saved in Xhip presets or banks nor VST presets. The control parameters are saved as part of the complete plug-in state in VST banks.

In addition the control section contains a knob used to adjust globally (for all instances) the number of unison waveforms used for the *super-saw* ramp oscillator unison. This parameter defaults to 16 and is never saved in any format or projects. It should not normally be used.

Glide mode

Depending upon the mode selected for **voices**;

In monophonic modes: **Normal**, **fingered** and **afingered**. In **normal** mode glide is always applied. **Fingered** mode is often also referred to as *legato* where glide is only applied between notes without a release occurring between them. **Afingered** is the opposite in which glide is only applied to notes with a release occurring between them.

In polyphonic modes: **Follow**, **retain** and **voice**. **Follow** sets the start point of the glide for new notes to the most recent note played. **Retain** starts new notes gliding from the current position of the last activated voice; when many notes are played at once they will all glide from the same source frequency. **Voice** mode allows each individual voice to retain its own frequency and upon receiving a new note-on will start to glide from that point.

Voices

This parameter provides **monophonic** modes including **lowest**, **highest**, **last** and **first** priorities as well as **polyphonic** modes from a single voice through to thirty-two voice polyphony.

Routing

Velocity and **bender** inputs can be routed to offset the preset parameter value. Velocity is applied individually to polyphonic voices while bender is *global*. **Scale** is the percentage of the range of the destination parameter to use. **Offset** allows an offset to be added to the parameter to avoid having to modify the setting of that parameter in the preset to suit the velocity/bender input.

Unison

This unison is a *true* voice unison. It activates a number of voices while applying detune to them. These are complete voices including all of the processing such as filters, waveshaper, envelopes and modulators. The CPU time required is exactly equal to that required for an equal number of voices. For example the voice counter will display 64 voices when using 16 voice polyphony and 4 voice unison.

Voices

The number of voices to use for unison from 1 to 16.

Detune

The amount of detune to apply between voices.

Random

The depth of random detune variation applied to each voice. Creates a pitch *humanization* effect.

Distribution

Ideally set to **0.5**, the power to which the detune fraction is raised. Given ^(1/2), the unison applies the square root of detune ratios which is guaranteed irrational excepting perfect squares 1, 4, 9 and 16.

The oscillator unison does not skip these perfect squares when computing detuning amounts. This makes the detune ratios somewhat less than ideal although this does not have any impact when using below nine voices.

If the detuning ratios between voices are irrational it guarantees that phasing between voices will never be able to *line up* with more than two voices in-phase at the same time. This will minimize amplitude and timbre variation due to phasing while using unison.

This parameter may be set anywhere from **1/8th** to **8**.

Flags

Use program-change

Accept program change messages via MIDI.

Immediate program-change

Immediately apply parameter changes to active voices upon receiving a program change message via MIDI or VST interfaces or user input via the GUI. If this option is not selected a voice will retain its settings until it is re-triggered or otherwise reactivated via voice stealing.

Filter program-change

Apply parameter smoothing upon parameter changes triggered via program change.

Drum mode

Drum mode allows the synthesizer to operate as multi-timbral such that it can produce multiple sounds at the same time from a single instance. It is most useful for percussion although quite limited as it only plays back presets at middle-c tuning and is not otherwise adjustable. Notes will be mapped to presets starting from **C2** to preset 1, **C#2** to preset 2 and so forth.

Mono re-trigger

Re-trigger the voice when pitch changes due to note-on in monophonic mode. This parameter has no effect in polyphonic mode.

High-pass

Enable a high-pass filter applied to the output of the synthesizer to eliminate sub-sonic content or *DC offset*.

File Formats

Xhip

The plug-in implements a custom preset format rather than depending entirely upon the host/VST interface functionality. Part of the reason for this is that Xhip existed long before it was wrapped in a VST interface.

Preset (.adxi)

Xhip presets contain the value of all of the main synthesizer parameters. Additional *global* settings from the control section are not stored in .adxi format presets.

Bank (.adxb)

The plug-in stores all of the 128 *user bank* presets in its .adxb format.

VST

The VST interface provides functionality via parameter and *chunk* get/set methods which allows the host to store both presets and banks as well as complete state information for project files.

The plug-in depends upon the host implementation for support of VST format files via the VST interface and does not support VST format files internally.

Preset (.fxp)

This functionality depends upon the host implementation, although in most cases .fxp format files contain the preset chunk provided by the plug-in. Xhip encapsulates the current preset in .adxi format in the preset chunk provided to the host.

Bank (.fxb)

Unfortunately VST does not distinguish between banks (a collection of presets) and the plug-in state saved in projects. Generally plug-ins will respond to a host request to save the bank chunk with the complete plug-in state. This allows the host to reload the complete state with project files.

The plug-in includes the MIDI CC map, currently selected background color and other options in this chunk along with the content as it would be saved in the internal .adxb format.

PCM

TODO: description of .wavs file list format.

TODO: describe Microsoft RIFF WAVE PCM format support, smpl and name chunks.

Xhip supports only 8-bit or 16-bit PCM format files. Multi-channel files are supported although only the first channel will be loaded with the others ignored.

PCM data can be loaded and used to replace the wave shapes built in to the oscillators. The data is output by using nearest-neighbour interpolation (zero-order hold) with the same anti-aliasing filter as used on the other waveforms.

Unfortunately PCM is not saved in presets, banks or projects (state, fxb) and functionality is extremely limited. It is not recommend that the PCM functionality should be used in projects where project recall is required.

In situations where project recall is not required the PCM functionality can be used to significantly extend the range of oscillator timbres which can be produced. This includes playback of fully anti-aliased sample content including for example drum sounds or vocals.

Other Functions

Init Bank

TODO: description of init bank functionality (Xhip format bank loaded from plug-in path.)

Preset Management

> *When I change some settings on a preset the changes will persist if I move to another preset and back. The only way I can get back to the original unmodified preset is by loading the original preset from a file. Is this is a bug?*

Not at all.

The advantage is that it isn't necessary to remember to *save* a preset as this occurs automatically as parameters are edited. If an *edit* or *working state* were used the preset would need to be saved manually to the *user bank* in order to ensure changes were maintained.

This would make complete control including parameter adjustments and program changes via an external MIDI controller impossible as there would be no way to save parameter adjustments without opening the GUI. Unfortunately MIDI does not include preset management functions such as save, load or init.

Xhip shuffles these issues around ever so slightly to make itself controllable via MIDI.

If a *permanent* save of a preset or bank is needed it must be manually saved to a file. Otherwise the entire *user bank* (in memory, per instance) is used as a *working state* and there should be no need to worry about forgetting to save changes before switching presets.

Tips & Tricks

TODO: feed me many pages of tasty sound-design ramblings ... nom nom nom

For the most part any "for example ..." or other paragraph which does not objectively specify properties or parameters of each section should be moved here.

Post-shaper Filter

The post-shaper filter can be used as a very simple EQ in either low, high or band/notch modes to pick out specific frequencies. While using the 12 dB high-pass mode tuned to **~52 Hz** the **Q** parameter can be adjusted to boost bass frequencies.

Another common application is to add overtones using the 12 dB modes with **Q** near 100%. The primary sound may be generated by the oscillators and main filter with a bell-like overtone added by the second filter acting if desired on harmonics added by the waveshaper.