# discoDSP HighLife


Users Manual

Open Source VSTi Sampler

http://www.discoDSP.com/

# Installation

To install HighLife, simply run the provided installation application. When prompted select your existing VST Plug-in folder on your hard disk.

# Description

HighLife is a performance sampler including VSTi sampling, sample editor, five built in effects, flexible modulation route and morphable scheme.

# Features

### Overview

- Built-in VST host for 32bit sampling of VSTis (freezing).
- Built-in wave editor.

- 32-Bit floating point based wavetables.
- 128 Programs with unlimited zones.
- Up to 32 voices polyphony (Multilayered zone sampling engine)

- Selectable interpolation engines: Hermite, Sinc 64 and Sinc 512.
- Both sinc modes are not suitable for realtime purposes but bouncing to audio.

- Morphable scheme and automatable zone parameters (incl. cue and loop points).
- Syncable LFO with configurable Phase.
- Stepping-free modulation sends, pitch bend and modulation to morphing wheels.

### Sampler Import formats

- Auto mapping when multiple files are opened at once.
- Samples: .wav (8 to 32 Bit), .mp3, .raw.
- Programs: propietary .fxp/.fxb, .akp (Akai S5000/S6000), SFZ.

### Sampler Export formats

- Samples: .wav.
- Programs: propietary .fxp/.fxb, SFZ.

### Program overview

- Unlimited zones.
- ADSR amplitude envelope
- ADSR envelope filter and pitch assignable (both bipolar).
- -24dB/Oct filter modes: Lowpass/Hipass/Bandpass/Notch and disabled.
- Stepping-free Cutoff and Resonance.
- High quality Chorus, delay and reverb
- Wow factor effects: Daft and rock da disco.
- Full mono/poly and legato operation.

### Zones overview

- Independent envelope amplitude, modulation envelope, LFO and filter settings.
- Automatable zone parameters (including cue and loop points).
- Up to 128 MIDI triggerable cues, with different start/end and loop settings each.
- Automatable per-cue pos/end/loop points, supporting swapping of cue range/loop
- markers.
- 4 Loop types (forward, bidirectional, backward, forward w/sustain).
- Group/Off by triggering settings.
- Amp Env, Mod Env and Lfo (syncable), 24dB Filter (5 Types)
- Full MIDI Input triggering control.
- Chorus, delay and reverb sends.
- Glide with auto-switch control.

### Sample editor overview

- Zoom up to 1:1 resolution.
- Clipboard: Cut, copy, paste and sample trim.
- Amplitude: Fade in/out, normalization and DC removal.
- Effects: Reverse, rectifier, sin/tahn drive and spectral mirroring.
- Cues and loops editing.

# Morphing

HighLife  layer contains two internal states: morph **source** and morph **sensibility**.  Each state contains a unique set of all values.  **Morphing** allows you to easily glide between these values.

The morph sensibility state is seen thru alpha blend, and can be modified by using CTRL key while adjusting the knob.

Once both states have been defined, you can easily morph between them by assigning the **Wheel MOD** or using standard MIDI velocity if **Vel > Mod** is assigned.


This picture shows a morphed knob state. The first value is located at eleven o'clock and morph state at two o'clock. It will result in a filter cutoff increase when using the mod wheel or MIDI velocity if assigned.

## 1.1 How to freeze

In order to freeze your favorite VSTi just follow these steps:

Click **VSTi button**, then search for a VSTi .dll. Once loaded you should see the VST name in the LCD display. In order to check the GUI to tweak any presets press **GUI button**.

You can access instanced plugin preset list by left clicking at the top part of the display LCD and HighLife preset list by left clicking at the bottom part. You can also find an alternative way by clicking the small navigation arrows located at the LCD top right.

Configure the sampling before pressing **Freeze** button:

- **Prgs**: Number of programs to be freezed. Depends of the instrument.
- **Layers**: Velocity layers divisions per freezed key.
- **Lenght**: Amount of seconds to be sampled of every key split.

- **Low**: Lowest key note to start freezing from.
- **High**: Highest key note to end freezing from.
- **Step**: Number of keyboard semitone increment scannings. Higher number results
- in less notes sampled. Ie.- 12 steps in a full scale results in one sample per octave.

Once you have freezed the VSTi you will be asked to uninstance it in order to save system resources. If for any reason you don’t want to do it you can leave it and use **Unload** button later on.

# 2. Auto mapping

HighLife allows .wav, .mp3 and .raw loading including auto mapping.

In order to get the auto-drum mapping, press the **Sample button** and once you are in the proper folder who contains the sampler you want to map use the standard Windows way to select samples. (by holding the mouse and selecting all the samples, or by using Ctrl and clicking individual samples).

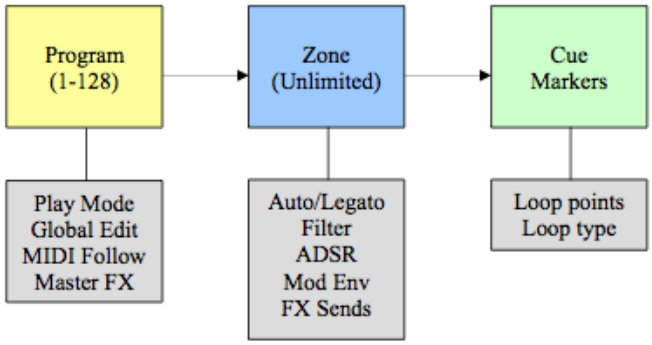
Once the samples are loaded the will be mapped in C4 or less if it the number exceeds the full octave.

You can change auto drum behavior under Options. Enabling it will map each sample into a different key. Disabling will result all the samples under the same root or based into WAV root key chunk.

# 3. Programs

## Overview

HighLife can manage up to 128 programs, with a unlimited amount of zones each. The architecture is designed as follows:




**Engine** settings will be applied for all the instances in the project at once in order to speed up the selection before the bounce process.

**Program Options**

- **Master volume**: Controls the amount of volume in the instance, starting at 0dB. This is a global setting.
- **Pitch Bend Range**, **Glide** and **Auto Glide** are zone related.
- **Pitch Bend Range**: The pitch bender allows you to drastically change the pitch of the current patch in real-time. You can modify the pitch bend range from 1 to 24 semitones.
- **Glide**: Glides the pitch between the current note and a newly played note. The higher the value, the slower the transition progresses.
- **Auto glide**: When this led is enabled (by clicking it), the glide/portamento will be only applied if overlapping notes are played (legato).

**Play Modes**

- **Mono**: Only 1 voice of the sampler is used for the layer. In this mode any note played will discontinue the previous note.
- **Legato**: A monophonic mode in which envelopes are not restarted when new keys are played.
- **Poly**: Polyphonic mode allows multiple notes to be played at once.

# 4. Zones

As you may have previously read, a HighLife program is based of an unlimited number of zones, each with the following settings marked in blank plus the ones under **Edit** which will be described in **section 4.5**:

## 4.1 Filter Section

Once the samples are mixed, sound is next routed through the HighLife filter section. This section will transform the frequency response of the signal.

A filter is a unit that changes the magnitude of a range of frequencies of the sound, boosting or cutting these frequency values.

### Basic Filter controls

The two most common filter parameters are:

- **Cutoff**: This value sets the frequency point affected by the filter response.
- **Q (aka Resonance)**: This value determines the amount of amplification of the range of frequencies surrounding the frequency (cutoff) point.
- **Key > Track**: Filtering applyed thru the keyboard based on the note position..
- **Vel > Track**: Filtering applyed thru the keyboard based on the MIDI velocity.

## Filter Types

HighLife has 4 filter types plus a disabled mode. You may toggle through these types using the leds located in the Filter area.

Available filter types follow:

- **BR**: The Notch filter will cut the frequency range surrounding the Cutoff point, and will be processed thru the 24dB Lowpass filter explained above, but using a slightly shifted Cutoff frequency. Using this with high Resonance amounts will result in interesting vocal formants effects.

- **LP**: A 4 pole Lowpass Filter with -24dB per octave rolloff. This functions similar to the LP 12db, but with a steeper frequency curve response.

- **BP**: A Bandpass Filter, which allows only the frequency range surrounding cutoff point to pass through. Resonance controls the size of this bandwidth.

- **HP**: The opposite of the Lowpass filters, the Highpass will pass frequencies above the cutoff point and will cut all range below. This filter type has a rolloff of -24dB per octave.

## 4.2 Envelopes Section

### Amplitude Envelope

Amplifier applies an envelope to the output gain of the current sound. The following controls apply:

- **Attack**: Controls for the time that envelope moves from note press (0) to full volume. (Does not apply to legato mode).
- **Decay**: This knob controls the time length between envelope gain goes from upper point (end of Attack) to Sustain level.
- **Sustain**: Sets the point at which the envelope stays while note is held. This state is held until the note is release.
- **Release**: Once the note is released, this knob will control the time in which the gain will fall from current envelope point (Sustain) into complete silence, finishing the voice use.

**Note:** Longer releases can result in many simultaneous voices, resulting in higher CPU usage.

- **Amount**: Adjusts the overall amount of the envelope.
- **Vel > Amp**: Adjust the overall volume of the instruments related to MIDI velocity.
- **Vel > Mod**: Adjust the overall morphing of the program related to MIDI velocity.

### Modulation Envelope

Modulation Envelope allows easy ADSR (**A**ttack, **D**ecay, **S**ustain, **R**elease) envelope based modulation of the filter Cutoff and Pitch control.

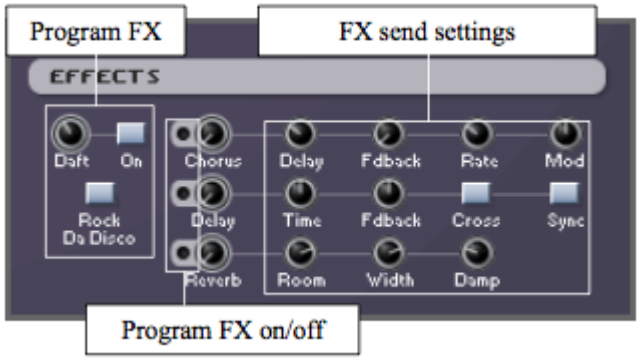
- **Attack**: Controls for the time in which the filter / pitch envelope moves from 0 (note play) to full filter (based on the Envelope Amount, below).
- **Decay**: Controls the time length between the end of attack and the sustain level.
- **Sustain**: Sets the filter / pitch point at which envelope stays while note is continually pressed, once attack and decay have completed.
- **Release**: Begins once the note is released, controlling the time the gain will fall from current envelope point (Sustain) until complete.
- **Envelope > Cutoff Amount:** The Envelope Amount option controls the amount of the filter envelope. The < > arrows means it's of a bipolar nature.
- **Envelope > Pitch Amount:** The Envelope Amount option controls the amount of the filter envelope. < > arrows means it's of a bipolar nature.

## 4.3 Modulation LFO

The modulation LFO allows to create more expressive sounds by routing the internal Low Frequency Oscillaor to four sends.

- **Phase**: Changes the value of the starting point of the LFO. If you want a **free running LFO** set it to max value.
- **Rate**: Velocity of the LFO. It can be synced to host tempo by clicking of the sync led.
- **Destinations**: The following destinations are available to the Modulation LFO:
    - **Amp**: Amplitude.
    - **Cutoff**: Filter cutoff. Bipolar value.
    - **Q**: Filter Resonance. Bipolar value.
    - **Pitch**: Global pitch by a range of **two** octaves. Bipolar value.

## 4.4 Effects




### Chorus

- **Level**: Amount of the chorus applied. Turning it to minimum values bypass it.
- **Delay**: Controls how much of the signal is captured and repeated.
- **Fbdack**: This controls how many times the signal is repeated.
- **Rate**: This controls the amount of the LFO.
- **Mod**: This controls the amplitude of the LFO.

### Delay

- **Level**: Amount of the delay applied. Turning it to minimum values bypass it. Delay can be **synced** or **switched to cross delay** by clicking the buttons in the section.
- **Delay**: Controls how much of the signal is captured and repeated
- **Fdback**: This controls how many times the delayed signal is repeated.

### Reverb

- **Level**: Amount of the reverb applied. Turning it to minimum values bypass it
- **Room** and **width**: This sets the length of time that the reverb lasts for - higher settings simulate larger spaces.
- **Damp**: Controls the rate at which reverberant energy is absorbed by the various surfaces in the environment.

### Wow-FX

These effects are program based.

- **Daft**: Special effect for those Daft sounds.
- **Rock Da Disco**: Instant sound puncher and pumper.

## 4.5 Zone Edit

### Overview

In order to configure the rest of the Zone parameters after importing samples or freezing, you may swith to **Editor mode**. If you have Program MIDI Zone Follow enabled, zones will switch automaticly to the triggered by MIDI notes.

You can also **select** a specific zone, **browse** samples for a specific zone, and **delete** current or all zones by clicking in the third general LCD row.

### Zone Edit properties

In the following area you can see and modify specific zone properties:

**Information properties**

- **Zone**: Name of the active zone.
- **File**: If available, path of the loaded zone.
- **Chan**: Waveform channels.
- **Rate**: Waveform sample rate.
- **Size**: Waveform size.

**Editable properties**

In order to edit the following properties you need to **left click** the **LCD**, **hold** and **move yourmouse up or down**.

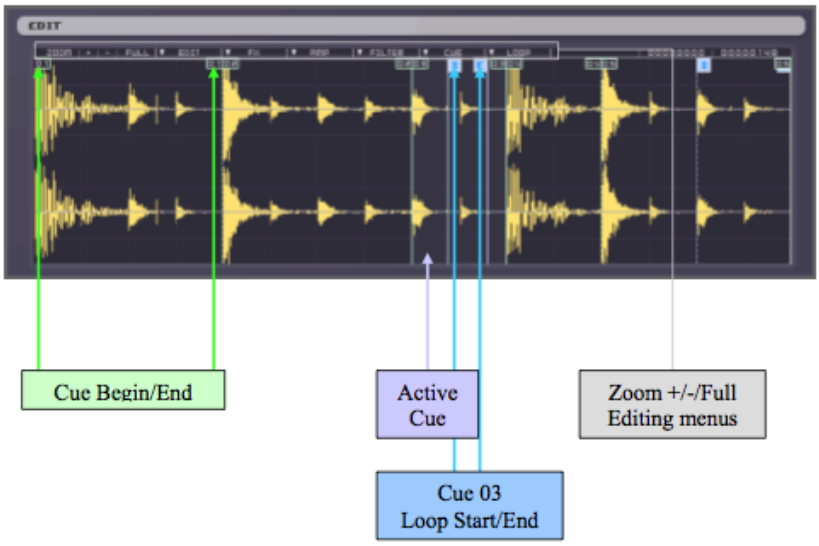
- **Gain**: Gain in dB.
- **Panning**: Panning.
- **Coar**: Transposition value in semitones.
- **Fine**: Transposition value in cents.
- **Kc**: Defines how much the pitch variates with every note. Default value is 100, which means pitch will variate one hundred cents (one semitone) per played note.

Setting this value to **zero** means that all notes in the region will play the same pitch, particularly useful when mapping **drum or sliced sounds**.

- **Group**: Exclusive group number.
- **Off**: Region off zone. When a new zone with a group number equal to offby plays, this zone will be turned off..
- **Tg:** Sets the trigger which will be used for the sample to play. Value can be either attack (plays on note-on), or release (plays on note-off).
- **Root**: Root note. It will be used as reference once multiple cues are setup.
- **Vel XXX to YYY**: Velocity range the zone will respond to.
- **Key XX to YY:** Note range the zone will respond to**.**
- **Sync:** Enables a loaded wave sync to host tempo. If a loop is loaded HighLife will set the number of ticks required to sync properly. This feature may be used with cue points as well.

## Sample editor

In addition to zone parameters, HighLife also comes with a built in sample editor able to do basic audio operations, several effects and cue/loop edition




### Editing tips

- Double clicking selects all the active sample window.
- Once the wave is zoomed, if you hold and drag the mouse from to the left or to the right you will be able to do scrolling with any of the
- In order to activate any looping, you must set a cue first, which will be played at the defined root note.
- For several cue removal you need to do a sample selection which covers the numbers you want to remove.
- Once **SYNC** button is enabled, doing the any sample selection with your mouse and **holding Ctrl** key will **snap to tick** the selection, which may be useful for quick looping and slicing selections.

# Version History

## Release 3

March 29, 2009

Greatly improved sfz import, more in line with the official sfz specs. Most sfz files can now be used rightaway without the need to edit out any unsupported features or moving the soundfiles around on the computer.

**SFZ Parser**

- Sample path can be relative or absolute to any location on the storage media:
- Path names may contain blank spaces
- <group> header can now be used.
- <key> opcode is now recognized.
- Comments can be put on the same line with opcodes.

**Fixes and changes**

- The following additional zone parameters can now be modified from the zone editor panel: root key, low key, high key, low velocity, high velocity.
- A couple of wave files that were refused by the previous version will now load fine.
- Loopinfo will not be used if loopstart > loopend (workaround for some nonstandard wav editors).
- When creating loop crossfades Highlife now implements a equal power crossfade instead of the previously used equal gain (linear) crossfade.
- Highlife won't crash anymore whenever a sample can't be opened. Instead an error message is displayed and the load process is reverted.
- Added Open Source Logo.
- About box. (Click on the Highlife Logo to open).

**HighLife SFZ Implementation**
Supported sfz Opcodes: sample, lokey, hikey, lovel, hivel, offset, tune, pitch_keytrack, pitch_keycenter, transpose, volume, pan, loop_start, loop_end, loop_mode, loccN, hiccN, lobend, hibend, lochanaft, hichanaft, lopolyaft, hipolyaft, trigger, group, off_by

**Supported Sfz Headers**
<region>, <group>

HighLife supports a subset of the sfz specification v1.0 opcodes. Unsupported opcodes can be included in the file. These will be ignored and should not cause any errors or misinterpretations. Unsupported headers (sfz v2.0 or third party) will be skipped. The parser will jump the next header if it finds an unknown one and ignore all opcodes inbetween.

## Release 2

March 16, 2008

George Yohng:

- Menu items added to save/load instrument in a special highlife format.
- Polyphony stealing is changed, so that the oldest voice is stolen
  (this seems to work better than what there was before).
- LFO fixed.
- Updates to make it compile well on MSVS 8.0.
- Changes to conform VST SDK better.

November 10, 2007

Chris Walton:

- Ported to VST SDK 2.4

## Release 1

Initial Open Source Release.

# Contact

http://www.discodsp.com/

# Copyright

Sfz definition file is © rgc:audio.

discoDSP HighLife is © by discoDSP.

## BSD License

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
- All advertising materials mentioning features or use of this software must display the following acknowledgement: This product includes software developed by discoDSP http://www.discodsp.com/ and contributors.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.