

HardSID MIDI v1.1

THE HARDSID MIDI SETUP UTILITY.....	4
COMMON SETTINGS	4
Current user	4
System clock	4
MIDI CONTROL PANEL (WINXP/2000).....	5
CHANNEL SETUP	5
Internal velocity controller	7
PATCH EDITOR.....	7
Patch structure.....	7
Load patch file	8
Unload patch file	8
Rename patch	8
Make default (for new parts)	8
Reset default part to original	9
LFO preview window	9
Export names to Cakewalk	9
Parts in the patch tree	10
Add new part	10
Deleting a part	10
LFOs in the patch tree	10
Add new LFO	10
Deleting a LFO	10
Patch settings.....	10
SID version selections	10
Play mode	11
Filter mode.....	11
LFO retriggering.....	11
Arpeggio mode	11
Init. Controllers.....	11
Info.....	12
Save button (next by the Info button)	12
Undo button (next by the Save button)	12
Part editor.....	12
Meaning of before release / after release	13
Oscillator.....	13
Common settings.....	13
Env. Init (ADSR envelope init. Time).....	13
Solo legato	13
Port. spd (Portamento speed)	14

Manual port. (Manual portamento).....	14
Arp. Spd (Arpeggio update time).....	14
PB down (Pitch bender - down).....	14
PB up (Pitch bender – up).....	14
Detune.....	14
Fix. Pitch (Fixed note pitch).....	14
Transp. (Transpose).....	14
Waveform	14
Vibrato	16
LFO.....	16
Depth.....	16
PulseWidth.....	16
PulseWidth envelope.....	16
PulseWidth LFO settings	17
LFO.....	17
Depth	17
Filter settings	17
Common settings.....	18
Filtered part.....	18
Filter mode.....	18
Filter envelope	18
C. target (Cutoff target).....	18
Cutoff controller	19
Filter LFO settings.....	19
LFO.....	19
Depth.....	19
Reso. (Filter Resonance)	19
Reso. LFO (Filter Resonance LFO).....	19
Volume envelope (ADSR)	19
Attack.....	19
Decay	20
Sustain.....	20
Release.....	20
How to set the individual values	20
At min velocity / At max velocity.....	20
Sync checkbox.....	20
ADSR Controllers	20
Table editor.....	20
What is the SID table editor	20
Entering values	21
Speed.....	21
Waveform	21
Gate.....	21
Ringmod	21
Sync	21
Test	22
FiltMode	22
CutOff.....	22
PulWth	22
Pitch	22
Loop back to step.....	22
Hold gate on table.....	22
Modify button.....	23
LFO editor	23
Base params (group).....	23
Target params (group)	23
Controllers (group)	24
Ctrl. LFO (Control LFO group)	24

Common settings	24
Name	24
Waveform	25
Ph. Offset (phase offset)	26
Cmb. LFO (combine with LFO)	26
Comb. (LFO combination method)	26
Invert	27
Abs (absolute value)	27
Params	28
Depth	28
Rate	28
Period	28
Pulsewidth (for Pulse LFO only!)	28
Interlace (for the interlace Comb. Method)	29

The HardSID MIDI Setup utility

Features:

- System settings
- Channel setup
- Patch creation/editing
- Exporting patch names to Cakewalk

Note for Win9x/ME users: The HardSID MIDI Setup utility isn't a MIDI sequencer/controller program. You need to use a MIDI sequencer to play with the MIDI driver (**Cakewalk**, **Cubase**, etc...). The Windows XP/2000 version can work as a stand-alone MIDI controller.



Common Settings

Current user

When you create a patch, this name will be saved with it, showing that you're the author of this patch.

If anyone downloads this patch from a patch database and loads it into a patch bank, the patch information will show your name.

Please enter your name here before creating patches!

System clock

Specifies the refresh rate for the HardSID MIDI. (20Hz..500Hz)

It's 200Hz by default. It means that everything will be updated 200 times in a second (four times faster than the average update frequency of old C64 tunes).

It also determines the average value of the HardSID MIDI latency. At the default 200Hz setting the latency will be around 2.5ms. You can go down even to around 1ms average latency if you raise the system clock up to 500Hz.

This value is pretty good on an average Celeron 300A.

If you don't need to modify this value, leave it on default for making the patch sharing easier between you and others!

MIDI Control Panel (WinXP/2000)

This feature is present in the Windows XP / Windows 2000 version only!



You surely know the annoying error message when a MIDI port is already allocated by another application. This won't happen with the HardSID MIDI, since the WinXP/Win2000 version of the HardSID MIDI Driver can handle multiple MIDI clients at the same time.

The MIDI Control panel uses this feature to connect the MIDI Setup Utility to your HardSID MIDI Synthesizer Driver without exclusively allocating its MIDI port.

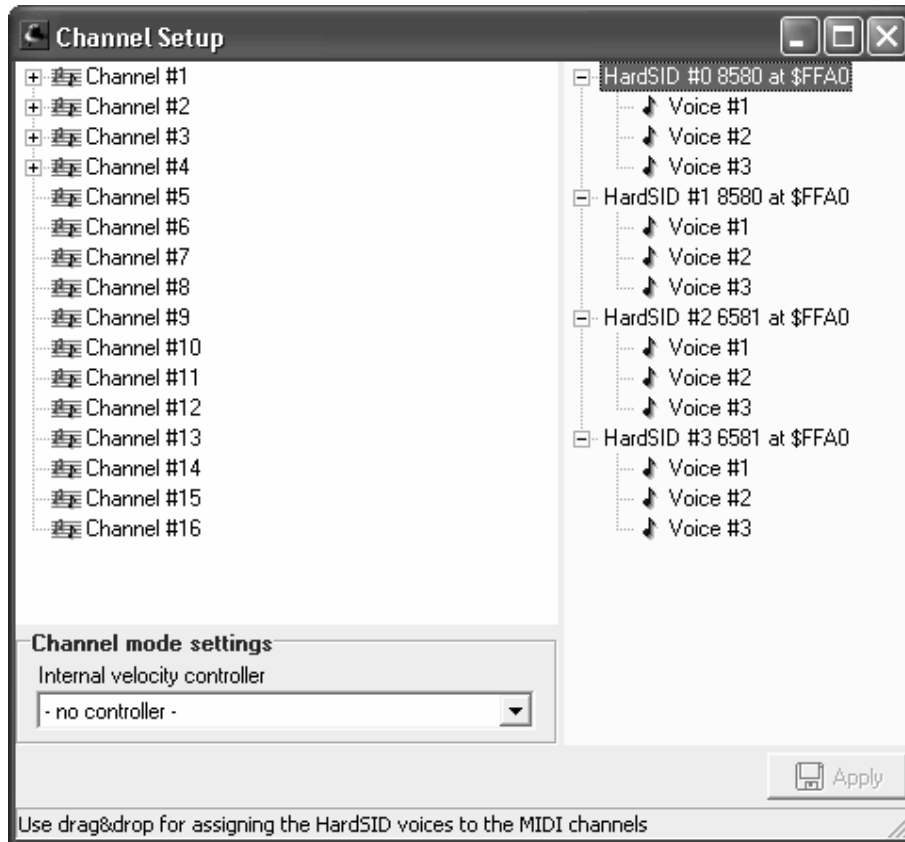
If you use a sequencer program (Cakewalk, Cubase, Logic, etc...), leave the "MIDI Control" checkbox unchecked or else both the sequencer and the HardSID MIDI Setup Utility will respond to incoming MIDI messages.

Turn the "MIDI Control" on if you don't use a sequencer program and you want to play with HardSID MIDI using your MIDI keyboard. You can select the MIDI input port you want to use from the "MIDI in" list. Select the MIDI channel to use from the "Channel" list or select "ALL" if you want to let your MIDI keyboard control the MIDI Channel selection.

Channel setup

The channel configuration is the heart of the HardSID MIDI.

You can make various channel settings using this window.



All of the HardSID devices are shown in the yellow device tree.

Every device has three voices, since they all contain a SID 6581/8580.

The Channel structure tree on the left contains the channel definitions for the HardSID MIDI driver.

You can drag & drop SID voices into the MIDI channels. One voice can be assigned to more channels, which gives you a flexible system.

When you play a Patch on a MIDI channel, the driver allocates the next **n** voices from the channel, where **n** is the number of parts within the patch.

Of course you can't play a patch with 3 parts on a channel which has only 2 SID voices. If you try this, you'll get no sound.

So, you need at least as many voices as many parts your patch has.

If you play a mono patch, the driver will always use the first voice of the channel.

Internal velocity controller

Any HardSID Patch can be controlled by controllers which can be defined in the Part settings and LFO settings.

How can you control your patches by the actual key velocity?

The answer is the Internal velocity controller. When you hit a key on your MIDI keyboard (or a sequencer sends note data), the HardSID MIDI driver sends the key velocity value to the MIDI channel using this controller number.

It means that you can use it in your patches like a simple MIDI controller.

Patch editor

The Patch editor lets you create/modify/load/unload HardSID patch files.
The HardSID MIDI patch bank contains 128 patch slots which can be filled with your patches.

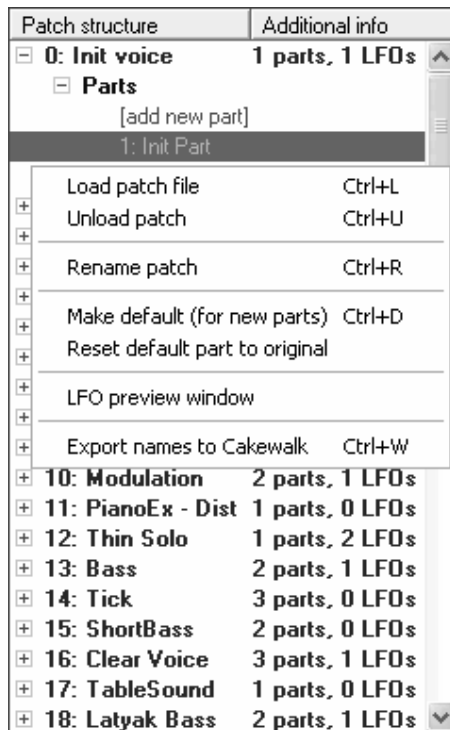
Patch structure

The Patch structure displays the patch information in your current setup. You can edit your patch bank and patches using the structure tree.

You can use drag&drop to copy a patch; part(s); LFO(s) to other patch; Parts section; LFOs section.

8 LFOs and 12 parts can be added to every Patch.

The HardSID MIDI driver can handle 32 LFOs at once, all of which can be fully routed within the patches.



Load patch file

Press Ctrl+L (or use the popup menu) if you want to load a patch file into the highlighted position. This patch can be accessed from MIDI sequencers (CakeWalk, CuBase, etc.) by its index (number). Use the 0..127 patch numbering style!

Unload patch file

Press Ctrl+U (or use the popup menu) if you want to unload a patch file from the highlighted position.

Rename patch

Press Ctrl+R (or use the popup menu) if you want to rename a patch at the highlighted position.

This function doesn't rename your patch file, only the patch definition within the file, since the actual patch name is stored inside the patch file.

Make default (for new parts)

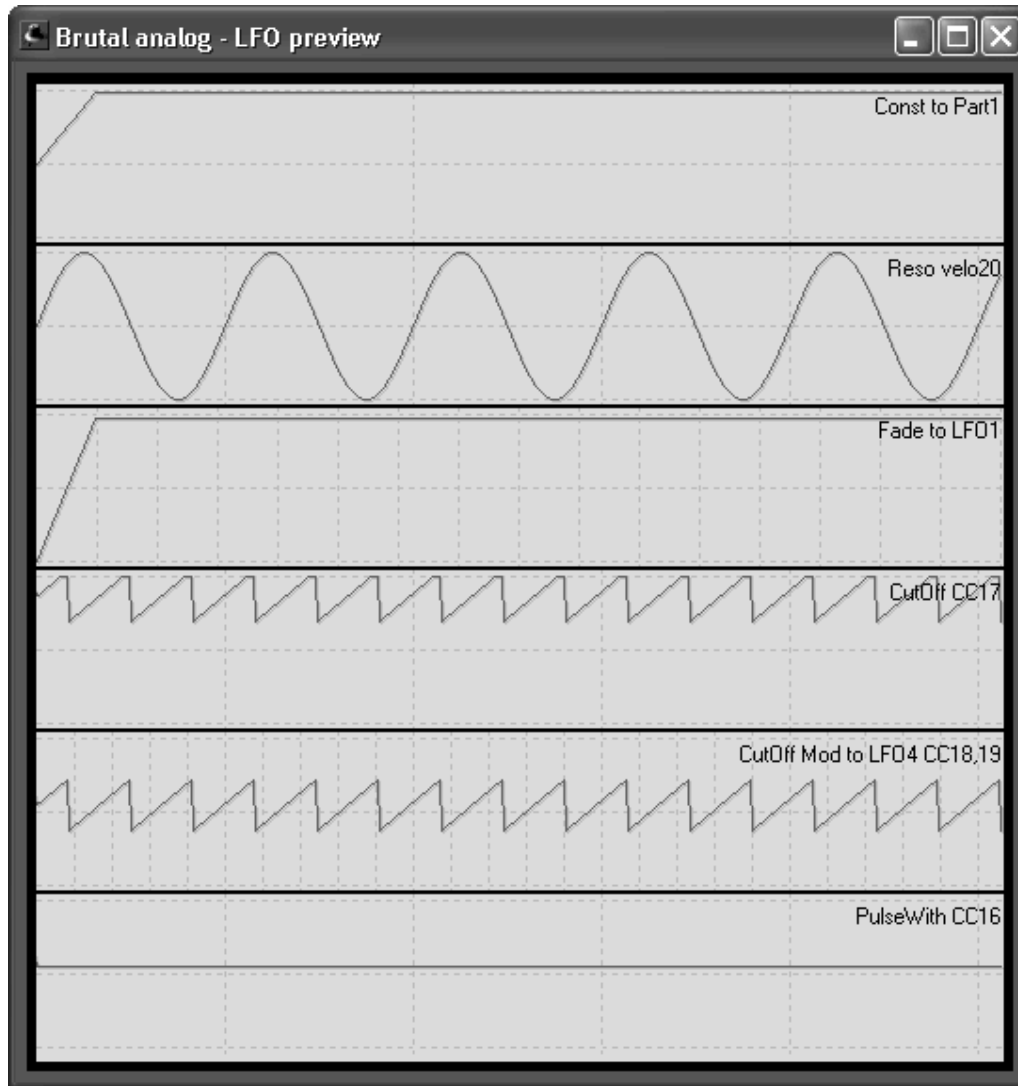
Press Ctrl+D (or use the popup menu) if you want to make the selected part become the default part definition. Once you've made a part default, every new part you add to patches will be created to be the same as the one you selected to be the default.

For example: You usually start with setting the waveform to triangle when creating a new part. This step can be skipped. Add a new part to a patch; set its waveform to triangle; "Make default". From this time, every newly added part will look like this part, so you won't have to set their waveform to triangle one by one.

Reset default part to original

The factory-default part becomes your default part.

LFO preview window



Displays the LFO definitions within the selected patch. This window is resizable and doesn't have to be closed and opened for every patch. You can simply open it once and roll over the Patch tree / edit your patches and the LFO preview will be updated on the fly.

You can set up patches using other LFOs and/or CCs (controller changes) in their definitions. The preview is recursive, so you'll see the effect one LFO makes to the other(s) on the fly. The CC values used for displaying the previews are coming from the controller initialization definitions. (learn more at Init. Controllers button)

Export names to Cakewalk

Press Ctrl + W (or use the popup menu) if you want to export all of your custom patch names to Cakewalk.

It's very useful if you don't like to see the standard MIDI patch names displayed all over Cakewalk instead of the real HardSID MIDI names.

Parts in the patch tree

12 parts can be added to every patch.

Add new part

Double click on the [add new part] line if you want to add a new part to a patch's structure. (max. 12 parts can be added)

Deleting a part

Simply press the Delete key on a highlighted part that you want to remove from the patch's structure.

LFOs in the patch tree

8 LFOs can be added to every patch.

Add new LFO

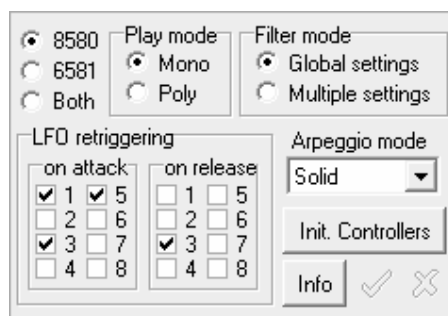
Double click on the [add new LFO] line if you want to add a new LFO to a patch's structure (max. 8 LFOs can be added).

Deleting a LFO

Simply press the Delete key on a highlighted LFO that you want to remove from the patch's structure.

Patch settings

You can view/edit the properties of the highlighted patch using this panel, which is situated under the patch structure tree.



SID version selections

This patch property was invented for administrative purposes. It can make future patch database engine project developments easier.

For example: If we make a patch database for www.hardsid.com, and you want to find bass patches for your HardSID which has a SID 8580, it'll be very easy by selecting the SID type from a list in the search engine.

- **8580** (this patch sounds correctly on the 8580 only)
- **6581** (this patch sounds correctly on the 6581 only)
- **Both** (this patch sounds correctly on both 6581 and 8580 versions)

This setting doesn't alter the sound of the patch, but it's very important for the future. Please, use it correctly!

Play mode

- **Mono:** Monophonic mode. The HardSID MIDI driver will always use the first n SID voices from the channel when playing a new note (where n is the number of parts in the current patch).
- **Poly:** Polyphonic mode, the HardSID MIDI driver will allocate new SID voices n times (for every part) from the channel when playing a new note (where n is the number of parts in the current patch).

Filter mode

- **Global Settings:** Every part will use the same filter setting within the patch. (This is useful for when the parts will use the same SID, depending on the number of SID chips, and the channel setup)
- **Multiple Settings:** Every part's filter settings can be different: it's used for high performance HardSID systems with 2 or more SID chips.

LFO retriggering

Check all of the LFO numbers which you want to be retriggered on note attack/release. Every patch's LFOs are numbered from 1 to 8.

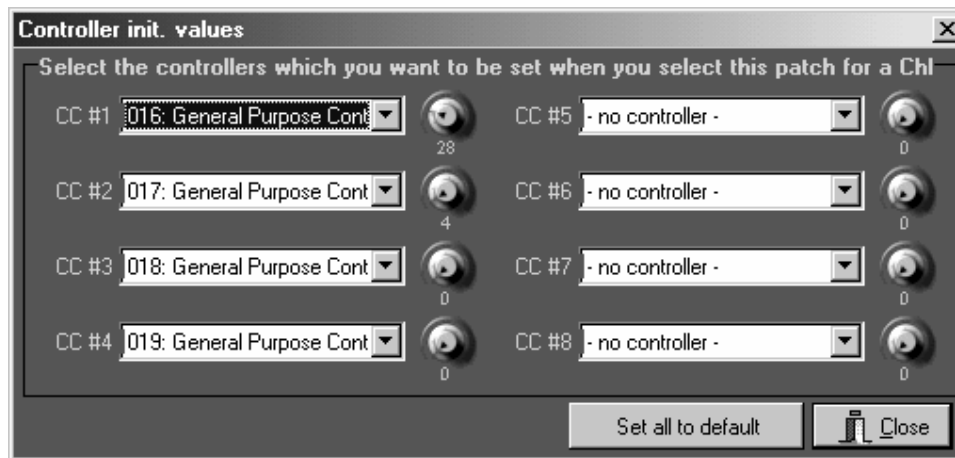
Arpeggio mode

- **None:** The arpeggio function is disabled for this patch.
- **Additive:** Notes can be only added to the arpeggio. Even if a note releases, it won't be removed until all of the notes are released.
- **Solid:** Notes can be added and removed to/from the arpeggio any time.

Init. Controllers

Controller initialization means that if this patch will be selected for a MIDI channel (Patch change occurs), all of the controllers which you selected for initialization will be set to the specified value.

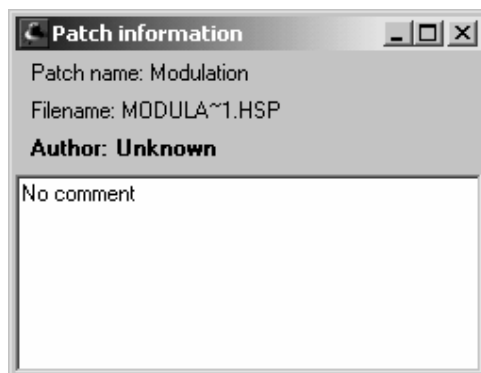
This is very useful for patches which can produce various sounds depending on CC (Controller Change) values.



Info

This window displays/edits the information associated with the highlighted patch file.

If you create a new patch, you can enter some information which will be shown to other users when they load this patch into their patch bank.



Save button (next by the Info button)

Press this button if you want to keep the editing you've made to your patch.

Undo button (next by the Save button)

Press this button if you want to cancel the editing you've made to you patch and return to the original state of your patch.

Part editor

This editor can be used for viewing/editing a part's properties. (You can use right-click on knobs which can select between a few distinct values, like: LFO selectors; LFO Combination selector; Filter/Resonance knob)

Meaning of before release / after release

You can see the 'before release' and 'after release' captions on many knob groups, and other places.

- **Before release:** These settings will be used while the note/key is still held down.
- **After release:** These settings will be used after a note/key is released.

Oscillator

This group encapsulates the part's common properties and all of the settings which can define a SID oscillator's output.



Common settings

Env. Init (ADSR envelope init. Time)

The volume envelope feature in the HardSID MIDI is based on the SID's (6581/8580) hardware ADSR (attack, decay, sustain, release) feature. This feature in the SID has a strange bug.

When you'll start creating patches, you'll probably notice that some parts' volume envelopes won't be stable at the beginning. You can fix this by setting the Env. Init to a higher value (20ms can solve this problem usually).

If a part sounds perfectly with the lowest setting, don't use a higher value, because this bug fix feature causes higher latency.

Solo legato

When the Solo Legato option is selected for a part, envelopes (ADSR, Filter, etc...), LFOs and some other parameters won't restart for a note which is started while the previous one is still held on the same channel.

This feature is used for smooth changes between notes instead of hardly restarting sounds.

* For MONO Patches only!

Port. spd (Portamento speed)

Specifies the time while the frequency of the last played note fades into the frequency of the actually starting note. When it is set to OFF, the portamento feature is disabled for the part.

The portamento feature is a very nice solo feature.

* For MONO Patches only!

Manual port. (Manual portamento)

When the Manual portamento is selected for a part, the portamento feature will be activated only for notes which are started while the previous note is still held on the same channel.

* For MONO Patches only! (with specified portamento speed)

Arp. Spd (Arpeggio update time)

Specifies the arpeggio update time (for patches which are using arpeggio).

Arpeggio update time: When the arpeggio update time elapses, the arpeggio engine takes the next active note from the arpeggio list.

PB down (Pitch bender - down)

Specifies the number of semitones which will be subtracted from the played note's semitone value when the Pitch bender is at its lowest position.

PB up (Pitch bender – up)

Specifies the number of semitones which will be added to the played note's semitone value when the Pitch bender is at its highest position.

Detune

Specifies the oscillator detune value.

- **Position = 0**: the oscillator is **not detuned**
- **Position = 63**: the oscillator is **detuned** by a whole **semitone (up)**
- **Position = -64**: the oscillator is **detuned** by a whole **semitone (down)**

Fix. Pitch (Fixed note pitch)

Specifies fixed semitone value of the part if it's set to a position other than Off.

Transp. (Transpose)

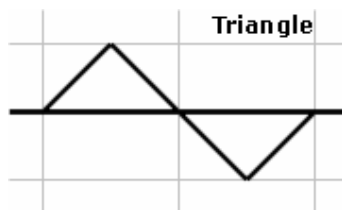
Specifies the amount of semitones which will be added to the played note's semitone value.

Waveform

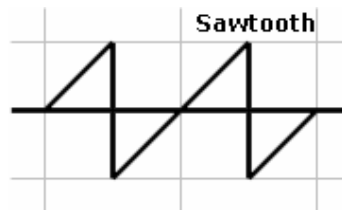
Specifies the oscillator's waveform, and the combination with another oscillator.

The following basic waveforms are available:

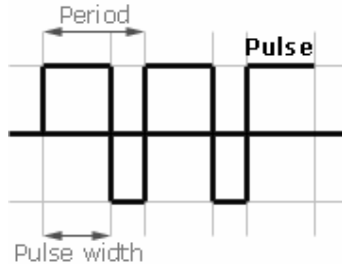
Triangle: A very smooth waveform, typically used for flute and whistle sounds.



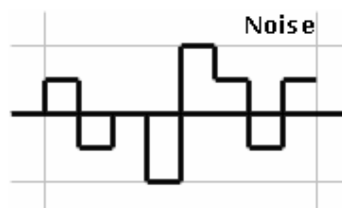
Sawtooth: A very sharp waveform, typically used for trumpet-like sounds.



Pulse: This waveform is very interesting, because its sound is variable using the pulse width value (oscillator group \ pulsewidth). Setting the pulse width to its center position results in a clear 50% square waveform, which was used, for example, in the Atari 8-bit computers. If you want a very thin sound, you can set it near the maximum or minimum value.



Noise: A random waveform, which is widely used for special effects, like drums, winds, and other interesting noises.



You can select more waveforms. The valid combinations depend on the SID version. The **6581 doesn't really support the mixed waveforms.**

There are many valid combinations for the **8580**. The only rule is that you **can't combine the Noise waveform** with any other waveform. All of the **other combinations are valid**.

Sync: Synchronizes this part's oscillator to the previous part's oscillator output. The following cases can happen, since a SID chip has only three oscillators:

1. #2 is synched to #1
2. #3 is synched to #2
3. #1 is synched to #3

What it means is when the synchronizer oscillator's value crosses zero (0), this part's oscillator restarts from the beginning.

Ringmod: Ring-modulates this part's oscillator value with the previous part's oscillator output (like described before).

What it mean is that the part's oscillator output will be the sum and difference of this oscillator's and the synchronizer oscillator's value.

Vibrato

You can assign the output of an LFO to modulate the actual oscillator frequency value.

LFO

Specifies the LFO which will modulate the oscillator frequency.

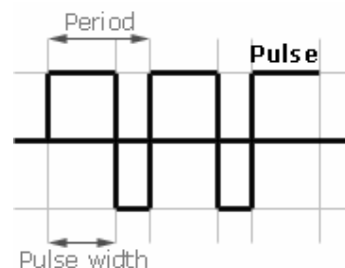
Depth

Specifies the multiplier factor for the assigned LFO.

The LFO's output value will be multiplied by this number before added to the oscillator frequency.

PulseWidth

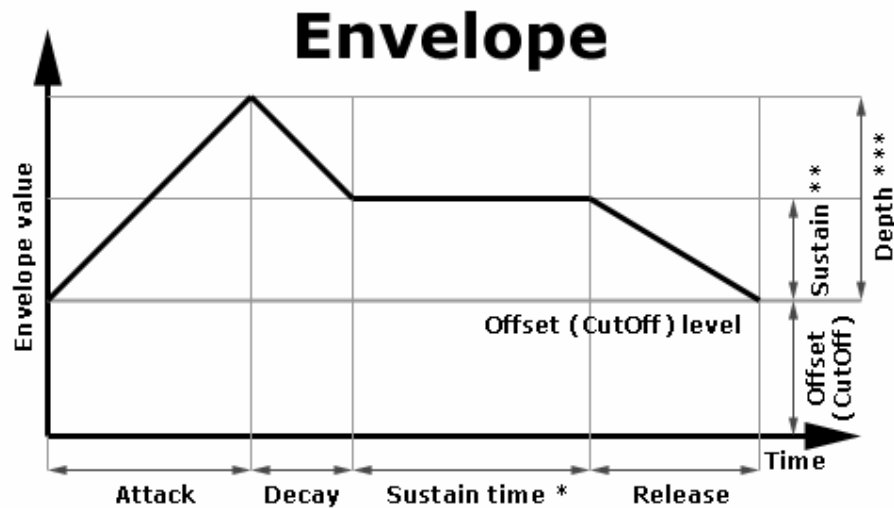
This feature sets the pulse width when the Pulse waveform is selected for the oscillator.



PulseWidth envelope

You can set up an envelope for the oscillator's pulse width.

When you set the Depth to zero (0), the envelope becomes a static pulse width value which is specified by the offset value.



* Sustain time: While the key is still pressed

** Sustain: It's specified compared to the depth value (%)

*** When the envelope is inverted, it's subtracted from the Offset (CutOff)

PulseWidth LFO settings

You can assign the output of an LFO to modulate the current pulsewidth value.

LFO

Specifies the LFO which will modulate the pulsewidth value.

Depth

Specifies the multiplier factor for the assigned LFO.

The LFO's output value will be multiplied by this number before added to the pulsewidth value.

Filter settings

The Filter settings group specifies all of a part's filter settings.



Common settings

Filtered part

Specifies whether the part is filtered or not.

Filter mode

Specifies the SID's filtering mode for this part.

Lowpass: Passes only those frequencies which are lower than the actual filter cutoff value.

Bandpass: Passes only those frequencies that are around the actual filter cutoff value.

Highpass: Passes only those frequencies which are higher than the actual filter cutoff value.

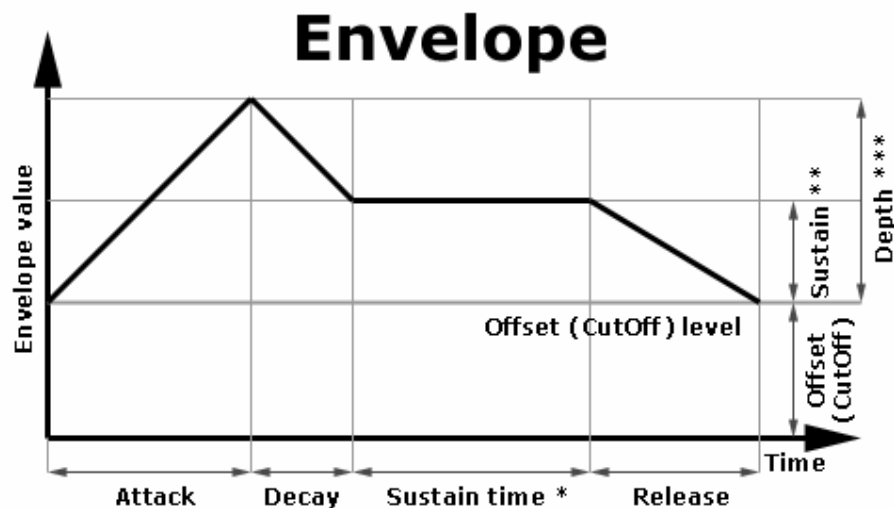
External: Enables filtering on the SID's external audio-in line.

Important note: If you'll play more parts on the same SID, you have to select the global filter settings option in the patch settings! If you try to use different filter settings on the same SID chip at the same time, it'll result in a random-like filter handling.

Filter envelope

You can set up an envelope for the filter.

When you set the Depth to zero (0), the envelope becomes a static filter value which is specified by the CutOff value.



* Sustain time: While the key is still pressed

** Sustain: It's specified compared to the depth value (%)

*** When the envelope is inverted, it's subtracted from the Offset (CutOff)

C. target (Cutoff target)

Specifies a second CutOff value which can be reached by sending a CutOff controller value 127 to the MIDI channel.

Cutoff controller

Specifies the CutOff controller, which can set the actual filter CutOff value between the original CutOff and the CutOff target value in 128 steps. (0..127)

To understand this, take a look at the Filter Envelope diagram's OffSet (CutOff) level axis. This axis can be moved from the original value to the target value using the CutOff controller.

Filter LFO settings

You can assign the output of an LFO to modulate the current filter value.

LFO

Specifies the LFO which will modulate the filter value.

Depth

Specifies the multiplier factor for the assigned LFO.

The LFO's output value will be multiplied by this number before added to the filter value.

Reso. (Filter Resonance)

Specifies the amount of resonance for the SID's hardware filter.

The resonance interval is 0..15, so it isn't a smooth SID feature.

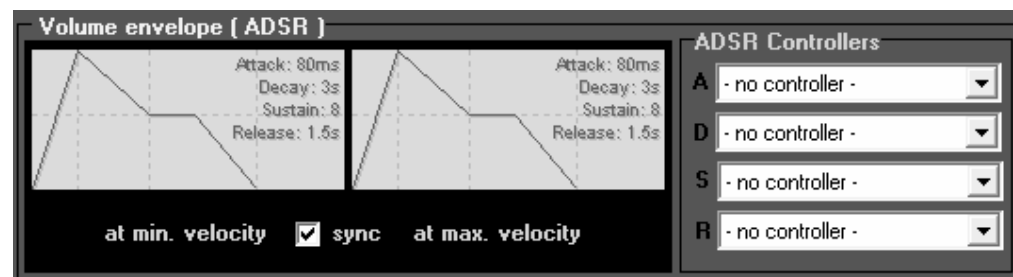
Reso. LFO (Filter Resonance LFO)

You can assign the output of an LFO to the current filter resonance value.

Note: If you assign an LFO to this property, it overrides the Reso. knob's setting.

Volume envelope (ADSR)

This group lets you set the part's volume envelope. This is called ADSR in the SID terminology (attack, decay, sustain, release).



Attack

Specifies how fast/slow the volume will reach the maximum level.

Decay

Specifies how fast/slow the volume will drop back to the sustain level.

Sustain

Specifies the sustain level.

The volume envelope will stay at this level while the key is still pressed (until the release state).

Release

Specifies how fast/slow the volume will drop back to zero (0) after leaving the sustain level.

How to set the individual values

Move the cursor over any of the bright dash-lines, and the cursor shape becomes a resizer tool. Now you can simply drag this line and move it in 16 steps to another position.

At min velocity / At max velocity

There are two equivalent ADSR editors side by side. You can specify an ADSR for the case when the key velocity is the lowest (0), and another one for the case when the key velocity is the highest (127).

The HardSID MIDI driver will interpolate every individual parameter (A, D, S and R) between the two specified values in 128 steps (0..127), depending on the actual key velocity value, or the assigned ADSR controllers.

Sync checkbox

It's not a run-time parameter. This checkbox makes the ADSR editing easier when you don't want to specify two different ADSR envelopes. If you want the max. Velocity synchronized automatically to the min. Velocity right when you adjust the min. Velocity, just check this checkbox. (This is checked by default, i.e. both envelopes are edited together.)

ADSR Controllers

Specifies the MIDI controllers for the A, D, S and R parameters of the volume envelope.

It overrides the actual key velocity when assigned. This way you can change each ADSR parameter's interpolation between the min. and max. value individually from a MIDI sequencer.

Table editor

What is the SID table editor

The SID table editor isn't an easy-to-understand feature for most musicians. Using this table requires a lot of SID programming experience.

What is this editor? There are many wonderful SID tunes which were written on the C64. Those tunes are using SID value sequencing for generating interesting drums and other special effects which can't be done just by adjusting some parameters. For example, a drum table plays white noises then pulse waveforms alternately with different filter parameters at very high speed.

Musicians will use the HardSID mainly for producing cool analog sounds, not for generating drums, because there are many great hardware and software samplers which can play real drum sounds.

Entering values

The following topics will show you exactly what you can enter into the cells of the table.

If you don't fill a cell, it'll inherit the actual value from the part definition.

Speed

Specifies the speed of the actual table step (row).

This value divides the HSMIDI system clock. For example, if the system clock is 200Hz, then speed 1 means that this step will be 5 milliseconds long. Speed 2 means that this step will be 10 milliseconds long, etc..

$$\text{StepTime(millisecs)} = 1000 / \text{HSSystemClockInHz} * \text{SpeedValue}$$

If you leave this cell empty and other columns in the same row are filled, the speed will be 1.

Waveform

Specifies the waveform of the actual step.

You can enter these characters into this cell:

- **t**: Triangle waveform
- **s**: Sawtooth waveform
- **p**: Pulse waveform
- **n**: Noise waveform

Gate

Specifies whether the Gate is enabled for this step. (When the Gate is not set, the volume envelope stays at the sustain level.)

You can enter YES or NO into this cell.

Ringmod

Enables or disables the oscillator ring modulation for this step.

You can enter YES or NO into this cell.

Sync

Enables or disables the oscillator synchronization for this step.

You can enter YES or NO into this cell.

Test

Enables or disables the Test bit for this step.

This is a very special bit. It was invented to initialize the oscillator generator when it hangs (the oscillator in the 6581 could hang when the noise waveform was selected together with another waveform).

You can enter YES or NO into this cell.

FiltMode

Specifies the filtering mode for this step.

The following characters can be entered into this cell:

- **l**: Lowpass
- **b**: Bandpass
- **h**: Highpass
- **x**: External filtering

CutOff

Specifies the filter cutoff value for this step.

Valid values: 0..254

PulWth

Specifies the pulse width value for this step.

Valid values: 0..254

Pitch

Specifies the fixed pitch OR the transpose value for this step.

Valid **transpose** values: **-64..+63**

Valid **fixed pitch** values: **0..126**

Loop back to step

The table can be looped back to any step.

If you enter zero (0), the table ends at its last value, and the sound continues with the actual part settings.

Hold gate on table

Specifies whether the default Gate handling is enabled or not.

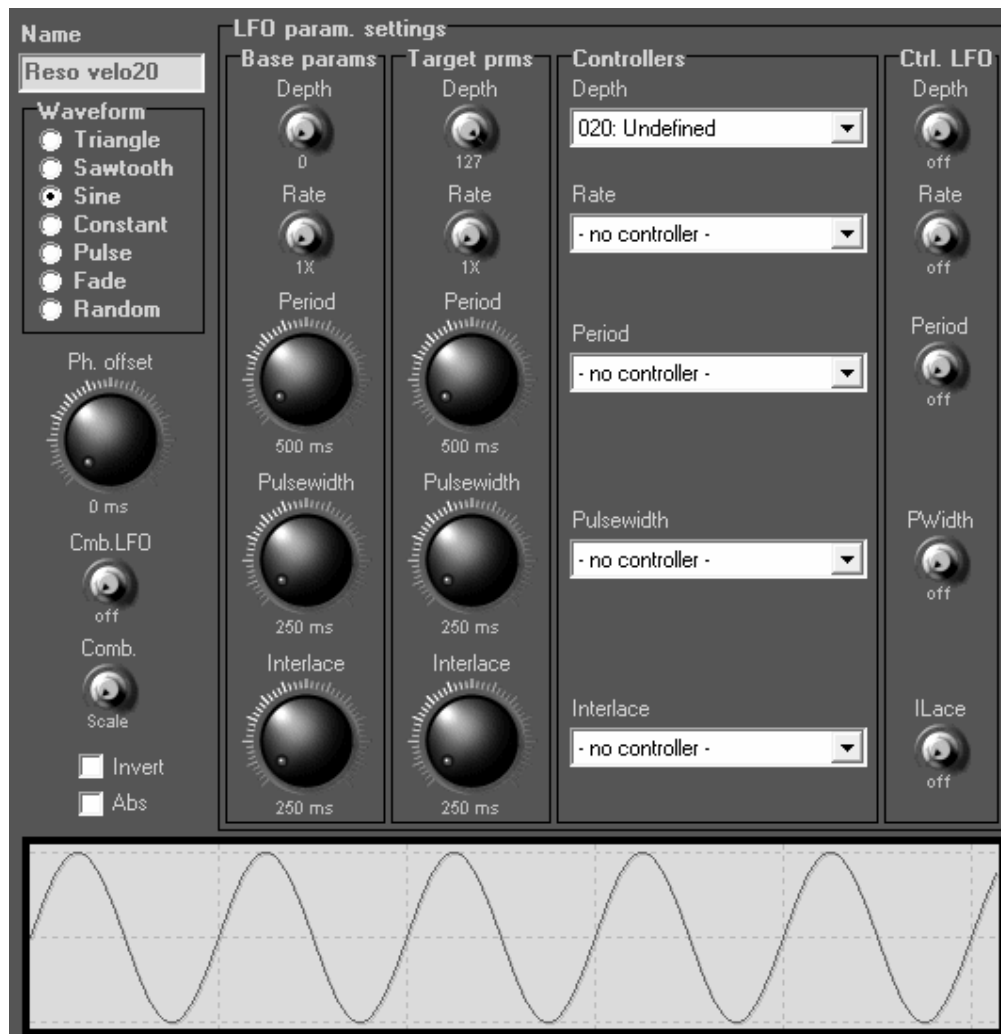
For example: If you created a drum voice, you don't want the driver to hold the ADSR sustain level while the key is pressed.

Modify button

If you want your settings to be entered into the part's definition, press this button!

LFO editor

You can edit LFOs using the built-in LFO editor of the HardSID MIDI. (You can use right-click on knobs which can select between a few distinct values, like: LFO selectors; LFO Combination selector; Filter/Resonance knob)



Base params (group)

These are the main parameters of the LFO.

Target params (group)

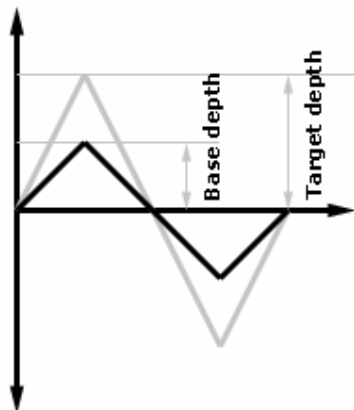
These are counterparts of the Base parameters.

They are used when there are parameter controllers assigned to the LFO. Parameter controllers can be MIDI Controllers (Controller group) or Controller LFOs (Ctrl. LFO group).

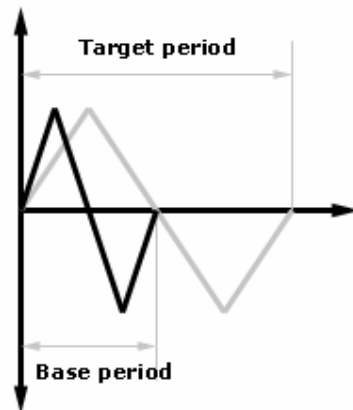
How does it work exactly:

Base/Target examples

Depth controlling example



Period controlling example



You can see the meaning of Base and Target parameters on these examples. The LFO engine calculates the current value between the Base and Target values using the actual parameter controller value (0..127: 128 steps).

For example: If Depth base = 10, and Depth target = 20., and the Depth controller value is 64, the current Depth value becomes 15.

Controllers (group)

You can assign MIDI Controllers to the LFO parameters.

If this controller changes, the LFO's actual parameter will change to a calculated value between the Base and Target values (see the Base/Target examples in the Target param help topic).

Ctrl. LFO (Control LFO group)

If you assign a Control LFO to a LFO parameter, the values of the control LFO will be handled like the previously described MIDI controllers (Controllers help topic).

These parameters will override the Controllers group's settings.

Common settings

Name

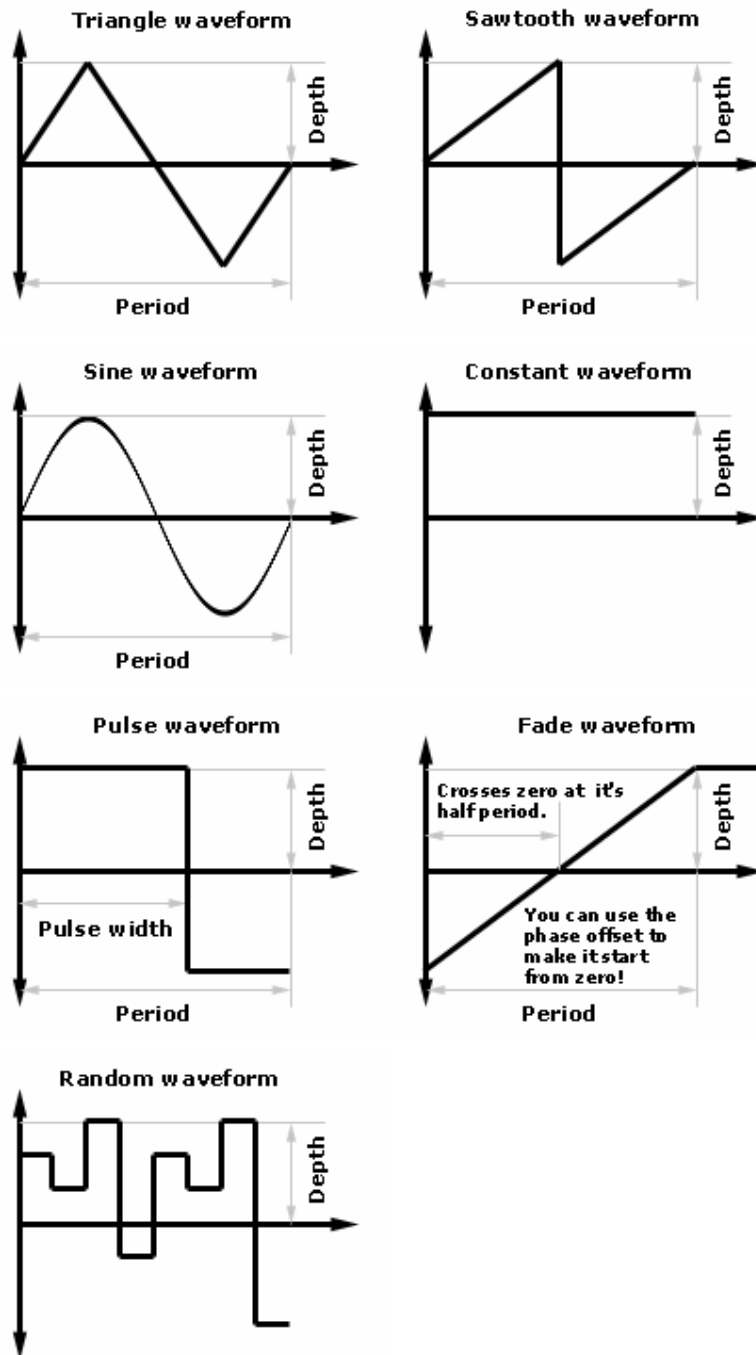
This is the name of the LFO.

You can see this name in the patch structure on the left side of the editor.

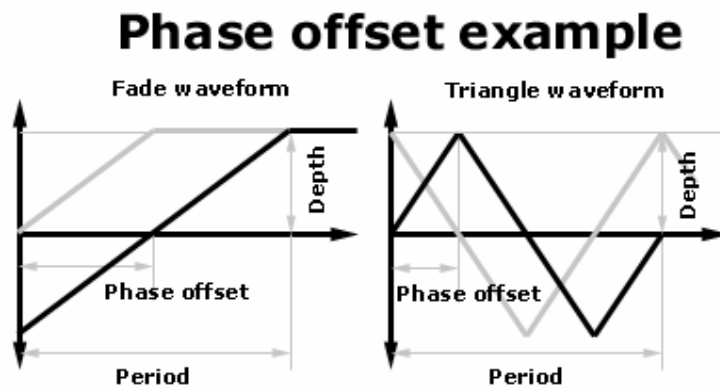
Waveform

This is probably the most important property of the LFO.

It's basically the same as a SID oscillator's waveform, but an LFO has more waveforms.



Ph. Offset (phase offset)



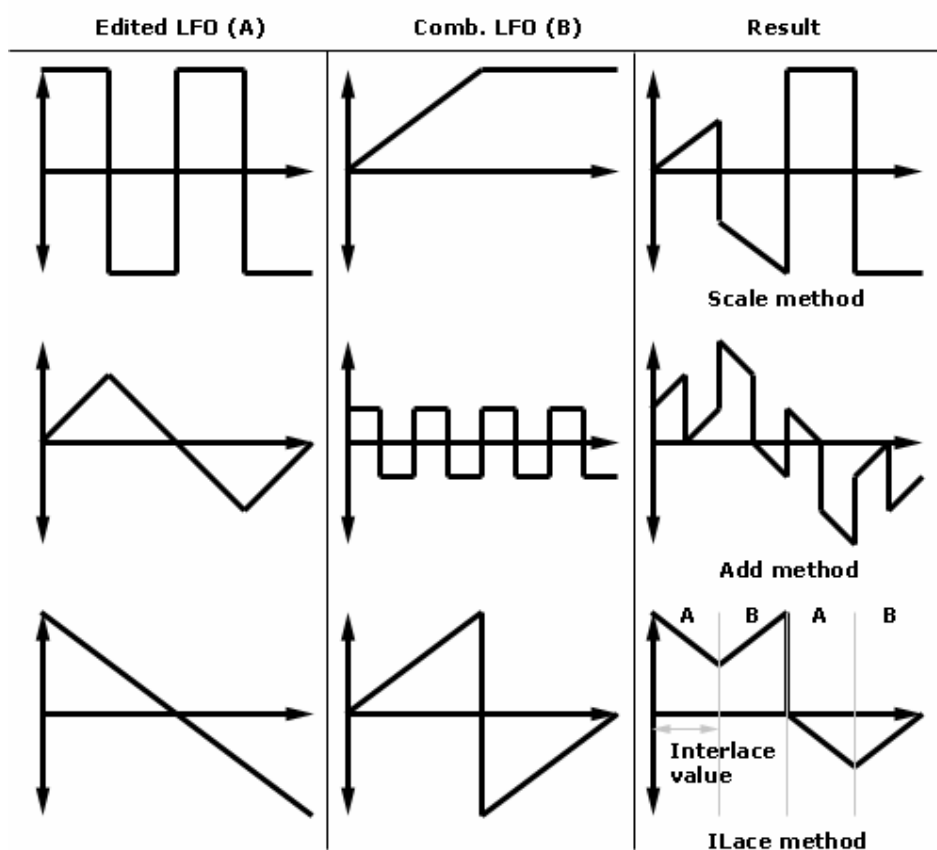
Cmb. LFO (combine with LFO)

You can combine the LFO with another LFO (1..8) within the same patch.
To understand this, see the Comb. (LFO combination method) topic!

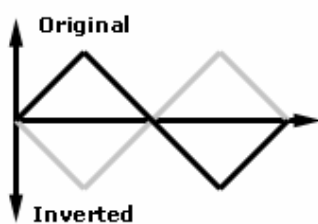
Comb. (LFO combination method)

Specifies the LFO combination method, when a Cmb. LFO is assigned.

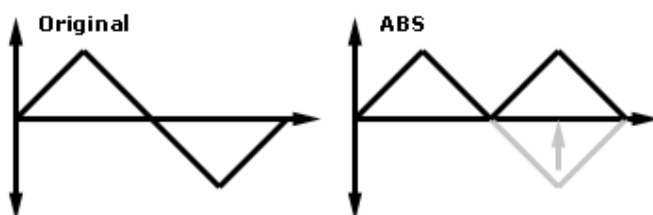
Comb. method example



Invert



Abs (absolute value)



Params

Depth

Specifies the LFO's depth.

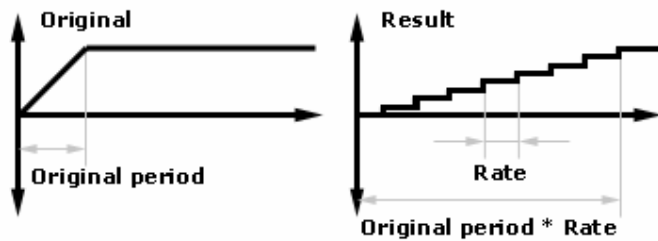
See the LFO waveform topic!

Rate

Specifies the LFO's Rate.

The result contains the same amount of LFO data as the original one, but every data is repeated Rate times.

This means that the new period will be the original period * Rate.



Period

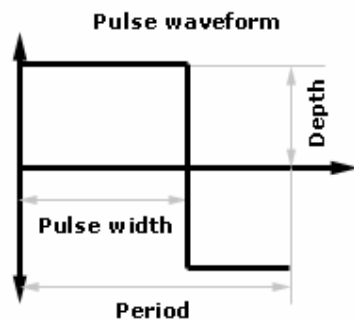
Specifies the LFO's period.

See the LFO waveform topic!

Pulsewidth (for Pulse LFO only!)

Specifies the pulse width of the pulse waveform.

The square waveform is a pulse waveform with 50% pulse width. ($\text{PulseWidth} * 2 = \text{Period}$)



Interlace (for the interlace Comb. Method)

Specifies the interlace period of the ILace combination method.

See the Comb. (LFO combination method) topic!