

the scrooo

Software Synthesizer Plug-In

Version 2.0

© 2011-2023 by Björn Arlt @ Full Bucket Music

<http://www.fullbucket.de/music>



VST is a trademark of Steinberg Media Technologies GmbH
Windows is a registered trademark of Microsoft Corporation
The Audio Units logo is a trademark of Apple Computer, Inc.

Table of Contents

| | |
|--|----|
| Introduction..... | 3 |
| What's New With Version 2.0?..... | 3 |
| Architecture..... | 4 |
| Oscillators..... | 4 |
| Formant Generators..... | 4 |
| Amplifiers..... | 5 |
| Modulation Sources..... | 6 |
| Envelopes..... | 6 |
| LFOs..... | 7 |
| Control Section..... | 8 |
| Options Menu..... | 8 |
| Voice Mode..... | 9 |
| LFO Phase..... | 9 |
| MIDI Learn..... | 9 |
| The scrooo.ini Configuration File..... | 9 |
| Parameters..... | 10 |
| Global..... | 10 |
| Oscillators..... | 10 |
| Formants..... | 11 |
| Amplifiers..... | 11 |
| Envelopes..... | 12 |
| LFOs..... | 12 |
| Frequently Asked Questions..... | 13 |

Introduction

The *scrooo* is a polyphonic software synthesizer plug-in for Microsoft Windows (VST2/VST3/CLAP) and Apple macOS (VST2/VST3/CLAP/AU) based on a spectral formant synthesis architecture and written in native C++ code for high performance. The main features are:

- Up to 64 voices polyphony including Unison mode portamento
- Two band-limited carrier oscillators
- Three fully independent formant generator sections
- Three formant generation modes
- Two amplifiers with volume and panorama control
- Three envelopes (ADSR) with exponential slopes
- Three low frequency oscillators (LFOs) with tempo synchronization
- MIDI Learn – all parameters can be controlled by MIDI CC
- MTS-ESP (<https://oddsound.com/>) dynamic micro-tuning support
- Plug-in supports Windows and macOS (32 bit and 64 bit)

The *scrooo* is based on the new **iPlug2** framework maintained by **Oli Larkin and the iPlug2 team**. Big thanks, guys!!! Without your work it would not have been possible to create a resizable *scrooo* user interface.

To resize the plug-in you just grab the yellow triangle at the bottom right of the window and drag it. You can save the current window size using the menu entry "Save Window Size" in the *Options Menu*.

If you have trouble with the standard version of the *scrooo*, please grab the (sound-wise identical) "N" version of the plug-in which is based on the original **iPlug** framework.

What's New With Version 2.0?

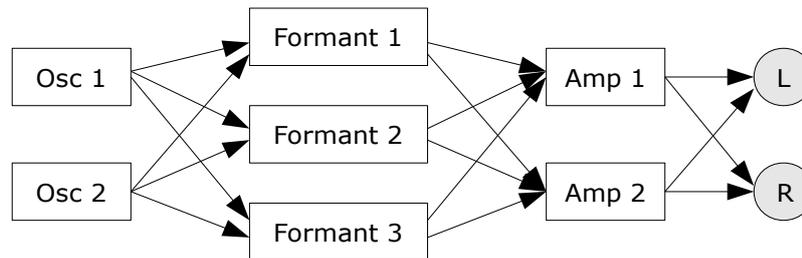
Version 2.0 of the *scrooo* is fully compatible with the previous version 1.x; it should be safe to replace the old plug-in with the new one.

The *scrooo* now supports MTS-ESP (<https://oddsound.com/>) dynamic micro-tuning.

Besides the new user interface I added two new features to the *scrooo*: *Unison* mode and optional *Single* trigger mode of the envelopes. I also did a lot of refactoring to the original code but I guess you will not notice much of it.

Architecture

Each voice of the *scrooo* features two oscillators feeding three separate formant



generators. The outputs of the formant generators are routed to two individual amplifiers with panorama control which are connected to the main stereo output bus.

Aside from the audio modules, the *scrooo* has three envelopes and three LFOs for modulation purposes.

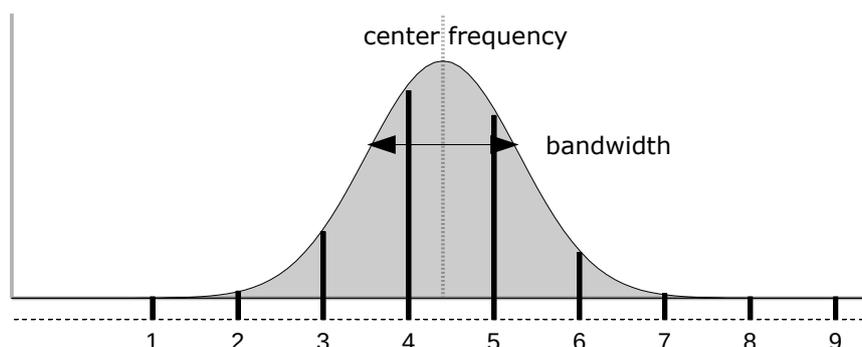
Oscillators

Each oscillator of the *scrooo* produces a sinusoid waveform feeding the three formant generators to create more complex signal spectra. The frequencies of both oscillators can be modulated independently by two different modulation sources (unipolar or bipolar).

On a new note event, digital oscillators typically start at the same initial phase angle of the waveform (e.g. the zero-crossing of a sine's rising slope) while classic analog oscillators do not; they are "running free" (which means they are still oscillating even if no note is played). The *scrooo* will emulate this behavior if the *Free* parameter in the Global section is turned on.

Formant Generators

The *scrooo* features three formant generators with individual center frequencies and bandwidths. Furthermore, the center frequency and amplitude of each formant can be modulated by various modulation sources.



Partial amplitudes for a given center frequency and bandwidth. The bell-shaped area shows the formant spectrum's "hull".

There are three modes of formant generation:

- **Fixed**
The center frequency of the formant is independent of the oscillator's base frequency; the generated spectrum will contain only integer multiples of the base frequency (i.e. harmonics).
- **Partial**
The center frequency of the formant is a fractional multiple of the oscillator's base frequency; the generated spectrum will contain only integer multiples of the base frequency (i.e. harmonics).
- **Ring**
The center frequency of the formant is a fractional multiple of the oscillator frequency; the generated spectrum typically will contain inharmonic frequencies.

Thus, *Fixed* and *Partial* modes result in the generation of a pure harmonic formant spectrum while *Ring* mode allows the creation of inharmonic spectra (similar to the output of a Ring modulator; hence the name). On the other hand, *Partial* and *Ring* modes generate formant spectra that are "shifted" along the frequency scale with respect to the base frequency while they are "fixed" in *Fixed* mode (which is a typical characteristic of "acoustic" instruments).

Each formant generator can be applied to only one or both of the oscillators; the resulting output can be sent individually to the two amplifiers.

Version 2 of the *scrooo* now generously provides an "oscilloscope screen" for each formant displaying rough sketches of the resulting wave form (*Partial* mode), wave shape (*Ring* mode) or frequency spectrum (*Fixed* mode).

Amplifiers

Both amplifier sections offer a stereo panorama control which can be modulated by any unipolar or bipolar source. The output volume is controlled by the *Level* parameter, the output signal of Envelope 1, and two optional unipolar modulation sources. Note that Envelope 1 is hard-wired to both amplifiers.

Modulation Sources

The *scrooo*'s oscillator and formant center frequencies, amplitudes etc. can be modulated by various modulation sources. Sources that produce a control signal of positive values only (including zero) are called *unipolar* while sources producing either positive or negative values are called *bipolar*. The following table lists the available modulation sources and their polarity.

| source | polarity | description |
|--------------|----------|--|
| <i>Off</i> | unipolar | constant value 0 |
| <i>On</i> | unipolar | constant value 1 |
| <i>LFO1+</i> | unipolar | output of LFO1 ranging from 0 to 1 |
| <i>LFO2+</i> | unipolar | output of LFO2 ranging from 0 to 1 |
| <i>LFO3+</i> | unipolar | output of LFO3 ranging from 0 to 1 |
| <i>Env1</i> | unipolar | output of envelope 1 ranging from 0 to 1 |
| <i>Env2</i> | unipolar | output of envelope 2 ranging from 0 to 1 |
| <i>Env3</i> | unipolar | output of envelope 3 ranging from 0 to 1 |
| <i>Velo</i> | unipolar | MIDI note velocity |
| <i>Note+</i> | unipolar | MIDI note value |
| <i>Rnd+</i> | unipolar | constant random value (for current note) |
| <i>PBnd+</i> | unipolar | MIDI pitch bend wheel (value at center is 0.5) |
| <i>Wheel</i> | unipolar | MIDI modulation wheel |
| <i>LFO1</i> | bipolar | output of LFO1 ranging from -1 to 1 |
| <i>LFO2</i> | bipolar | output of LFO2 ranging from -1 to 1 |
| <i>LFO3</i> | bipolar | output of LFO3 ranging from -1 to 1 |
| <i>Note</i> | bipolar | MIDI note (value at C3 is 0) |
| <i>Rnd</i> | bipolar | constant random value (for current note) |
| <i>PBend</i> | bipolar | MIDI pitch bend wheel (value at center is 0) |

Envelopes

The three envelopes of the *scrooo* are standard ADSR generators with exponential slopes as featured in classic analog synthesizers. Envelopes 2 and 3 also feature an initial delay parameter. Envelope 1 is hard-wired to the amplifiers and directly controls the overall amplitude contour. It is possible to switch from *Multiple* to *Single* trigger mode using the designated *Trig* switch.

LFOs

The three low frequency oscillators (LFOs) generate a periodic control signal from 0 to 100 Hz and/or can be tempo-synchronized to the host. Seven waveforms are available: *Sine*, *Triangle*, *Square*, *Saw Up* (rising sawtooth), *Saw Down* (falling sawtooth), *S/H* (Sample and Hold, i.e. random values), and *3-Step* ("stair"-shaped waveform with three steps). The *Retrig* parameter controls whether the LFO is restarted for each new note or is "running free" (similar to the Free Run mode of the oscillators).

It is possible to modulate the magnitude of the LFO's output by any unipolar modulation source (even by the LFO itself). This can be used to control the LFO via the modulation wheel or to create complex modulation signals, "magic vibrato" effects, etc.

In *Unison* mode you also can set the phase offset of the stacked voices – see section LFO Phase.

Control Section

Besides selecting programs, setting the *Free* parameter (see section *Oscillators*) and the *Voice Mode* as well as *Pitch Bend* intensity and *Portamento* time, this section of the *scrooo* provides some handy utility functions.

Options Menu

When clicking on the *MENU* button, a context menu opens with the following options:

| | |
|---------------------------------------|--|
| Copy Program | Copy current program to internal clipboard |
| Paste Program | Paste internal clipboard to current program |
| Init Program | Initialize the current program |
| Load Program | Load a program file containing a patch to the <i>scrooo's</i> current program |
| Save Program | Save the <i>scrooo's</i> current program to a program file |
| Load Bank | Load a bank file containing 64 patches into the <i>scrooo</i> |
| Save Bank | Save the <i>scrooo's</i> 64 patches to a bank file |
| Select Startup Bank | Select the bank file that should always be loaded when the <i>scrooo</i> is started |
| Load Startup Bank | Load the Startup bank file; can also be used to check what the current Startup bank is |
| Unselect Startup Bank | Unselect the current Startup bank |
| Default Path for Program Files | Sets the default path for program and bank files |
| MIDI Thru | Set globally if MIDI data sent to the <i>scrooo</i> should be sent through to its MIDI output |
| Ignore Program Change | Set globally if MIDI Program Change data sent to the <i>scrooo</i> should be ignored |
| Reload Configuration | Reload the <i>scrooo's</i> configuration file (see section <i>The scrooo.ini Configuration File</i>) |
| Save Configuration | Save the <i>scrooo's</i> configuration file (see section <i>The scrooo.ini Configuration File</i>) |
| Check Online for Update | When connected to the Internet, this function will check if a newer version of the <i>scrooo</i> is available at fullbucket.de |
| Window Size... | Change the window size of the <i>scrooo</i> |
| Save Window Size | Stores the current window size to the configuration file so that it will be restored next time you load the <i>scrooo</i> |
| Visit fullbucket.de | Open fullbucket.de in your standard browser |

Voice Mode

The number of voices in standard *Poly* mode ranges from 1 to 64. The new *Unison* mode allows up to 8 slightly detuned voices stacked on top of each other for various polyphonic modes. Clicking on the digital display opens a menu where you can select the desired *Unison/Poly* mode.

LFO Phase

In *Unison* mode, the *LFO Phase* parameter controls the “phase offset” of the LFOs per stacked voice. This means that for any value of *LFO Phase* greater than zero the LFOs of each stacked voice start at a different position in the wave cycle. Using this feature you can create great swirling pads or funky effects.

Note that *LFO Phase* does not have any effect in standard *Poly* mode.

MIDI Learn

Every parameter of the *scrooo* can be controlled by one MIDI controller. If you want to change the assignment of MIDI controller (CC; *MIDI Control Change*) to *scrooo* parameter the *MIDI Learn* function comes in quite handy: Just click the *Learn* button on the *scrooo*'s control panel (caption turns red) and wiggle both the MIDI controller and the parameter you want to assign (you can abort *Learn* by clicking the red button again). To save the controller assignments use “Save Configuration” in the *Options* menu (see previous section).

If you want to unlearn the assignment, right-click the *Learn* button (the label now reads “*Unlearn*”) and activate it. Now wiggle the MIDI controller or the parameter that you want to unlearn.

The scrooo.ini Configuration File

The *scrooo* is able to read some settings from a configuration file (*scrooo.ini*). The exact location of this file depends on your operating system and will be displayed when you click on “Reload” or “Save Configuration”.

Parameters

Global

| parameter | description |
|------------------|---|
| <i>Voices</i> | number of polyphonic voices (up to 32) |
| <i>Unison</i> | Switches from <i>Poly</i> to <i>Unison</i> mode |
| <i>LFO Phase</i> | Phase offset of the LFOs in <i>Unison</i> mode |
| <i>PBend</i> | maximum pitch bend amount (± 24 semitones) of the oscillators |
| <i>Porta</i> | portamento time (0 to 5 seconds) |
| <i>Volume</i> | the overall volume |
| <i>Free</i> | controls whether the oscillators are restarted or are "running free" (like classic analog oscillators) if a new note event occurs |
| <i>Clip</i> | controls if the overall output is unlimited or clipped to unity Note: The clipping LED will light as soon as the signal clips; thus, you can use <i>Clip</i> to create a distortion effect while attenuating the overall output using the <i>Volume</i> control. |

Oscillators

| parameter | description |
|-------------------------|---|
| <i>Pitch</i> | the relative pitch (± 24 semitones) |
| <i>Tune</i> | fine tuning of the oscillator (± 1 semitone) |
| <i>FM1 / FM2</i> | amount of frequency modulation |
| <i>FM1 / FM2 Source</i> | the source of the frequency modulation; this can be any modulation source (see <i>Modulation Sources</i> above) |

Formants

| parameter | description |
|---------------------------|--|
| <i>Mode</i> | <p><i>Fixed:</i> the center frequency is fixed i.e. independent of the base frequency; only harmonics of the base frequency are generated</p> <p><i>Partial:</i> the center frequency is a fractional multiple of the base frequency; only harmonics of the base frequency are generated</p> <p><i>Ring:</i> the center frequency is a fractional multiple of the base frequency; typically, inharmonics of the base frequency are generated</p> |
| <i>Range</i> | center frequency of the formant |
| <i>Spread</i> | bandwidth of the formant |
| <i>FM1 / FM2</i> | amount of center frequency modulation |
| <i>FM1 / FM2 Source</i> | the source of the center frequency modulation; this can be any modulation source (see <i>Modulation Sources</i> above) |
| <i>Volume</i> | initial amplitude of the formant |
| <i>Osc 1 / Osc 2</i> | enable / disable formant generation for Oscillator 1 and 2 |
| <i>Osc 1 / Osc 2 Send</i> | balance of formant output sent to Amplifiers 1 and 2 |
| <i>AM</i> | amount of amplitude modulation |
| <i>AM Source</i> | the source of the amplitude modulation; this can be any unipolar modulation source (see <i>Modulation Sources</i> above) |

Amplifiers

| parameter | description |
|-------------------------|--|
| <i>Pan</i> | stereo position (panorama) |
| <i>Pan Mod.</i> | amount of panorama modulation |
| <i>Pan Source</i> | the source of the panorama modulation; this can be any modulation source (see <i>Modulation Sources</i> above) |
| <i>AM1 / AM2</i> | amount of amplitude modulation |
| <i>AM1 / AM2 Source</i> | the source of the amplitude modulation; this can be any unipolar modulation source (see <i>Modulation Sources</i> above) |
| <i>Level</i> | output level |

Envelopes

| parameter | description |
|----------------|---|
| <i>Single</i> | switches from <i>Multiple</i> to <i>Single</i> trigger mode |
| <i>Delay</i> | initial delay time in seconds (Envelopes 2 and 3 only) |
| <i>Attack</i> | attack time |
| <i>Decay</i> | decay time |
| <i>Sustain</i> | sustain level |
| <i>Release</i> | release time |

LFOs

| parameter | description |
|------------------|--|
| <i>Waveform</i> | seven types are available: <i>Sine</i> , <i>Triangle</i> , <i>Square</i> , <i>Saw Up</i> (rising sawtooth), <i>Saw Down</i> (falling sawtooth), <i>S/H</i> (Sample and Hold, i.e. random values), and <i>3-Step</i> ("stair"-shaped waveform with three steps) |
| <i>Retrig</i> | when a new note event occurs, <i>Retrig</i> controls whether the LFO starts at the beginning of the wave form or is "running free" (similar to the <i>Free</i> control for the audio oscillators) |
| <i>Rate</i> | rate or speed of the LFO (in Hertz or note lengths) |
| <i>Sync</i> | controls whether the LFO is synchronized to the host tempo |
| <i>AM</i> | amount of amplitude (output level) modulation |
| <i>AM Source</i> | the source of the amplitude modulation; this can be any unipolar modulation source (see <i>Modulation Sources</i> below) |

Frequently Asked Questions

How do I install the scrooo (Windows VST2 32 bit version)?

Just copy the files `scrooo.dll` from the ZIP archive you have downloaded to your system's or favorite DAW's VST2 plug-in folder. Your DAW should automatically register the *Bucket ONE* VST2 plug-in the next time you start it.

How do I install the scrooo (Windows VST2 64 bit version)?

Just copy the file `scrooo64.dll` from the ZIP archive you have downloaded to your system's or favorite DAW's VST2 plug-in folder. Your DAW should automatically register the *Bucket ONE* VST2 plug-in the next time you start it.

Note: You may have to remove any existing (32 bit) `scrooo.dll` from your VST2 plug-in folder or else your DAW may screw the versions up...

How do I install the scrooo (Windows VST3 64 bit version)?

Just copy the files `scrooo64.vst3` from the ZIP archive you have downloaded to your system's or favorite DAW's VST3 plug-in folder. Your DAW should automatically register the *Bucket ONE* VST3 plug-in the next time you start it.

How do I install the scrooo (Mac VST2/VST3/AU 64 bit)?

Locate the downloaded PKG package file `scrooo_2_0_2_mac.pkg` in Finder (!) and do a right- or control-click on it. In the context menu, click on "Open". You will be asked if you really want to install the package because it comes from an "unidentified developer" (me ☺). Click "OK" and follow the installation instructions.

What is this "Formant Synthesis" thing?

There are two major approaches to waveform generation: *Subtractive Synthesis*, where a complex waveform is *reduced* to a less complex one using filters, and *Additive Synthesis*, where a complex waveform is *constructed* using a large number of simple (sinusoid) waveforms¹. *Formant Synthesis* (as implemented in the *scrooo*) is more on the latter side but works slightly different: A "carrier" waveform is created from a sinusoid using a non-linear transform (*Wave Shaping*) and "tuned" to a specific frequency (the center frequency of the formant). Thus, the spectrum of the formant is generated by shifting the (more or less controllable) spectrum of the carrier waveform to the formant's frequency domain. Since the *scrooo* supports the addition of up to three formants (or even up to six if the oscillators and formants are adjusted in a twisted way) the overall spectrum can get quite complex.

If you want to learn more I highly recommend a visit at <http://www.puredata.org/> . Here you can download the *Pure Data* application, a great audio (and graphics) processing tool including many examples and background information.

1 Apart from that, many other paradigms such as *Frequency* or *Phase Modulation* etc. exist.

Hey, no filters in the scrooo?

Yes, that's true, there is no filter inside the *scrooo*... well, to be honest there is a DC blocker right before the overall output but that doesn't count, right?

No filters? Then it's just a piece of crap!

If you do think so, I strongly recommend not to install the *scrooo*. If not, you may ask yourself...

What is the plug-in ID of the scrooo?

The ID is 3 3 1 5 .

How can I decrease the scrooo's CPU load?

Always try this:

- If you don't need a formant to be generated for one or both oscillators, switch off the respective *Osc 1* or *Osc 2* button of the formant.

Whenever it does not degrade the *sound* you need, try this:

- Reduce polyphony, i.e. the number of voices.
- Turn of Free Run mode of the oscillators.
- Turn of tempo-synchronization of the LFOs.

How can I prevent output clipping?

Decrease the formants' and/or the amplifiers' *Level* parameters. Alternatively, switch on the *Clip* parameter in the Global section but that might lead to distortion effects. Hey, maybe that's exactly what you need? ;-)

How can I fine-tune the Parameters in the scrooo editor? Especially the Range knob...?

Press the *Shift* key while moving a knob to increase it's resolution.

Hint: If you double click on a knob, the corresponding parameter will be reset to it's default value.