chipsynth PortaFM

Manual v1.105

Introduction

We are quite excited to finally share with you the first member of our new *chipsynth* line!
The research behind this new line started where we left off with chipsounds (2009), and has continued ever since. Everything we have learned using new hardware analysis rigs and software tools built in house culminated into this very moment.

But we also wanted our first chipsynth to cover synthesis methods we have touched before. So even though we have been working on much improved C64's and 2A03 models, an FM synthesizer seemed like the obvious missing link in our offering.

But since we are never going along the current we are pushing back our DX7 (6OP) and our MegaDrive(Genesis) (4OP) for a measly - in appearances - 2OP Portable FM synthesizer.

I always had a love for very crude digital sound generation. Stuff that is unique and gritty. The unique 'lofi' quality of the OPLL was never recreated this accurately before, we think.

Oh and it uses ***no*** samples for the FM generation at all!
We could easily have done that and spent 100x less time and resources doing so, but that's neither who we are as a company and what motivates us...

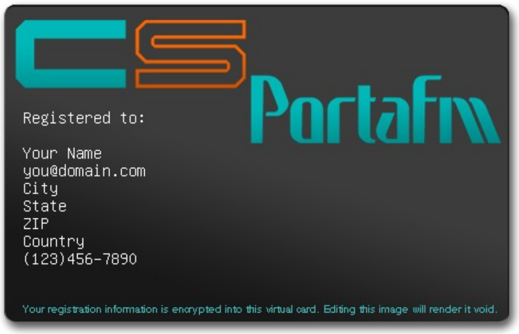
We want to understand how things really tick, and that doesn't come easy.

David Viens
May 15th 2018

Authorizing chipsynth PortaFM

You need to authorize PortaFM to make it fully functional, otherwise it will run in DEMO mode. When you order a license from our web store, you receive a personal Activation Key card named xxxx_key_1100.png as an email.

The Activation Key card is an image resembling a typical credit card.
This image contains your registration and details encoded within the Key card image. It will look like this:



Registered to:

Your Name
you@domain.com
City
State
ZIP
Country
(123)456-7890

Your registration information is encrypted into this virtual card. Editing this image will render it void.

You should save the xxxx_key_1100.png image file to your hard drive (keep it in a safe place).
For convenience, we recommend that you initially save the .png file to your desktop.
You will also receive a copy of the license card in your email inbox.

1) Locate the “license card” image where you saved it on your hard drive. (**xxxx_key_1100.png**)

2) Open the chipsynth PortaFM software application, or launch your favorite host and make sure you see the PortaFM interface.

3) Simply click and hold on the file, drag the “license image” or file icon directly onto the application's UI itself, and release it.

If you don't get any message (or are not able to drop the key in that host), try to import the png file from the the snapshot load menu or the slot load ***import***. (use *.* as file filter)

You should be presented with a message saying “Plogue Art et Technologie, Inc chipsynth PortaFM is now activated for {your name here}”

If none of those methods work, please contact chipsynth.support@plogue.com and attach your key.

Extremely Important!!

The xxxx_***key_1100.png*** file contains your ***sensitive personal information***, encrypted inside, including your **full name and address** taken from the online shop. Carefully protect this file. DO NOT GIVE THIS FILE TO ANYONE OR DISTRIBUTE IT IN ANY WAY OR YOUR PERSONAL INFORMATION WILL BE COMPROMISED. IF THE FILE BECOMES PUBLIC THE CARD NUMBER WILL BE BLACKLISTED AND THE CARD REVOKED. WE ARE NOT RESPONSIBLE IF YOU GIVE YOUR PERSONAL DETAILS TO A THIRD PARTY. IF THE CARD IS STOLEN, CONTACT US IMMEDIATELY. Without a valid card you will also not be able to obtain critical updates to the program.

Important Note:

If you have special circumstances or require site licensing, please contact us.

Updating to the Latest Version

Be sure to check the Plogue Web sites for any possible updates that have occurred since the time your version of the software was released. Software is frequently updated and a more recent version may be available.

Getting Help

The first place to look for a solution to any problem you may be experiencing is in this manual. Please read the manual before contacting support. Next, check the readme files (if any) which contain important information and all last-minute changes that haven't been available when creating this guide.

PortaFM is dynamic — evolving and growing. Please check the support area of our website at www.Plogue.com for the latest up-to-date information products, troubleshooting, FAQs, helpful hints and tutorials. Another resource is the support forums.

Whenever you encounter problems, you should also check if you have installed the latest updates. The version number of your software is displayed in the **About** dialog. Updates are released regularly to fix known problems and to improve the software.

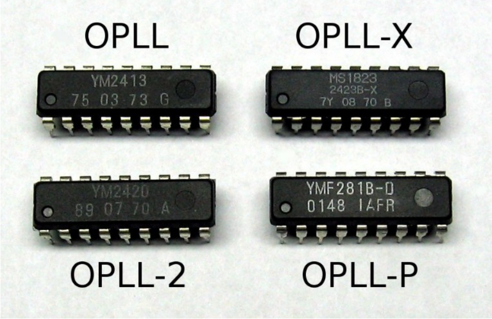
If you can't find a solution to your problem please email us at chipsynth.support@plogue.com. The best way to get the help you need is by giving us plenty of detailed information about the problem you are having. We do ask you to read this guide thoroughly and exhaust the other avenues of support before contacting us.

The Plogue forum can be accessed at: http://www.plogue.com/phpBB3/
You don't have to register to browse posts, but before you can post, you will have to sign up.

End User License Agreement : See Licence.rtf, which contains the full license agreement.

Plogue Art et Technologie, Inc.
www.plogue.com
www.facebook.com/wwwPLOGUEcom
twitter.com/wwwPLOGUEcom

Meet the OPLL



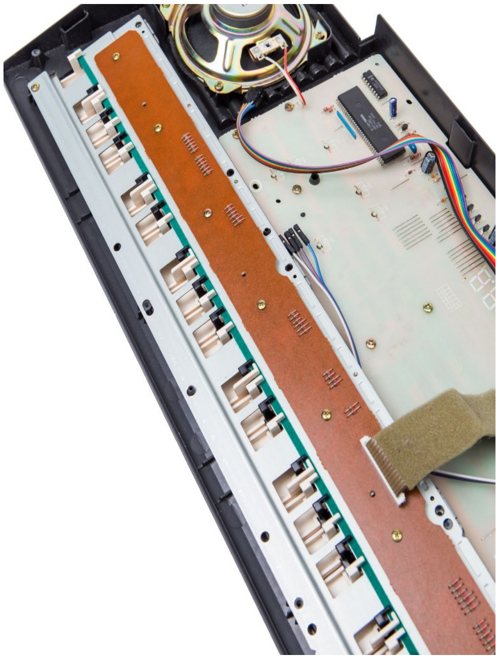
Plogue chipsynth PortaFM features the OPLL: the smallest, simplest synthesizer chip in the history of FM synthesizers.

Following the massive success of the DX7, Yamaha wisely decided to produce less expensive versions of its designs for the home and OEM markets, and produced chips with simplified calculations, all the while adding multitimbrality. Their new smart multiplexed designs made it possible to pack lots of features in very small packages, since the same processing blocks could be cycled between voices.

The OPLL is a cost-reduced version of the OPL chip: most of the instrument patches and drums are built-in, and it has a built-in DAC, making it very cheap to produce and to use.

This allowed the OPLL to be used in very simple systems. It was initially designed for Teletype systems such as the Japanese Captain, but the low cost and ease of integration has allowed it to be used in a wide variety of computer-based systems:

- Entry level digital keyboards such as the Yamaha PortaSound line: PSR-6, PSS-170/270, and the PSS-140 and SHS-10 (as the OPLL-2 variant).
- MSX-Music, used on the MSX computer, initially introduced as the FM-PAC cartridge, and then built into some MSX2+ machines and the MSX TurboR.
- The FM Sound Unit add-on to the Sega Mark III (Japanese version of the Sega Master System).
- Konami's Famicom game Lagrange Point (using the VRC7 mapper, which includes a cut down version of the OPLL with different built-in patches).
- The OPLL-X variant (different patches) is used in the FM Melody Maker cartridge for Atari ST.
- Various 16-bit Arcade systems.



Yamaha PSS keyboard. Notice how sparse the main board is.

PortaFM emulates the OPLL with an extremely high attention to detail. Voice mechanics are replicated down to the single bit, including exact oscillator waveform generation and modulation, envelope step patterns, DAC voltage imperfections, single sample delays between oscillators and even behavior in rare corner cases. The built-in OPLL patches are also recreated with ultra-high accuracy.

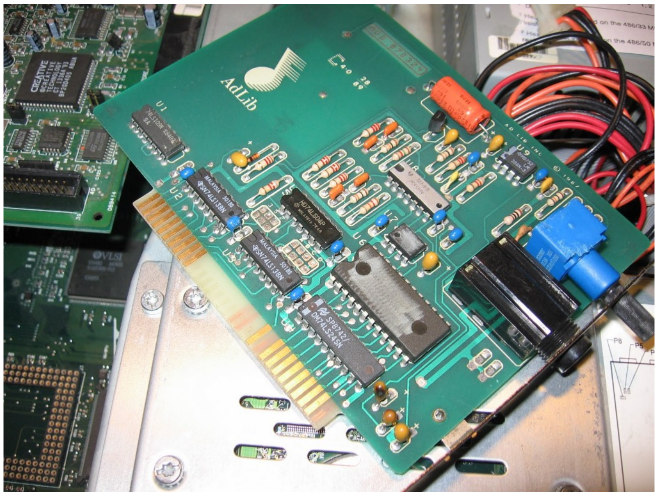
Having this level of precision in the emulation might seem overkill, but it contributes a lot to making every note unique, which happens in the OPLL due to envelope stepping changes depending on which exact cycle notes trigger on, and the special behavior of envelopes on key on where the time to reach the attack peak depends on the release state of the channel.

OPL2 : OPLL's big brother

New to the version 1.080 update, we've added our lovingly modeled OPL2 emulation. This chip is very culturally important to us, and we feel like we did justice to it.

The original OPL chip was Yamaha's first stab at a low cost FM chip designed for the home market. It had a fairly clean signal path compared to other chips of its time, with the pioneering idea of generating sound at a 49khz sample rate through a surprisingly clean DAC. It was used in:

- Arcade games such as SNK's Athena
- The Sound Expander cartridge for the Commodore 64
- MSX-Audio cartridges on MSX computers (in the form of the Y8950 chip, which added an ADPCM playback channel)

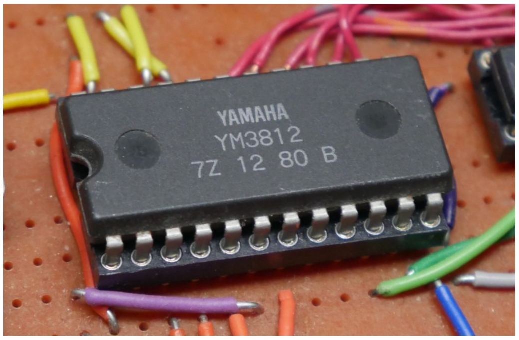


AdLib soundcard. The chip ID (Yamaha Japan YM3812) was sanded off to prevent cloning. This did not help!

The OPL2 is an early update of this chip, adding only a single thing: the ability to use the half-sine, abs-sine and pulse-sine waveforms in addition to pure sine waves. It is otherwise exactly identical. The added selectable waveforms add a surprising versatility to the available timbres. It was used in a wide array of systems:

- The AdLib sound card for the IBM PC. This was the first reasonably-priced sound card that offered a substantial improvement over the PC speaker, and this use of the OPL2 chip was quickly replicated by other sound cards such as the Sound Blaster, forming a de facto standard that was used all over multimedia PCs and in countless games.
- Home keyboards such as the Yamaha PSS-460.
- Arcade games such as Raiden.

As with all other of our emulations, our recreation of the OPL2 is extremely detailed, and any note will be produced in bit-perfect accuracy. We hope that this will let you appreciate this classic video game sound in its full glory.



OPL2 in its Plogue-reseach habitat

"FM1312" - just a sticker over the chip ID!

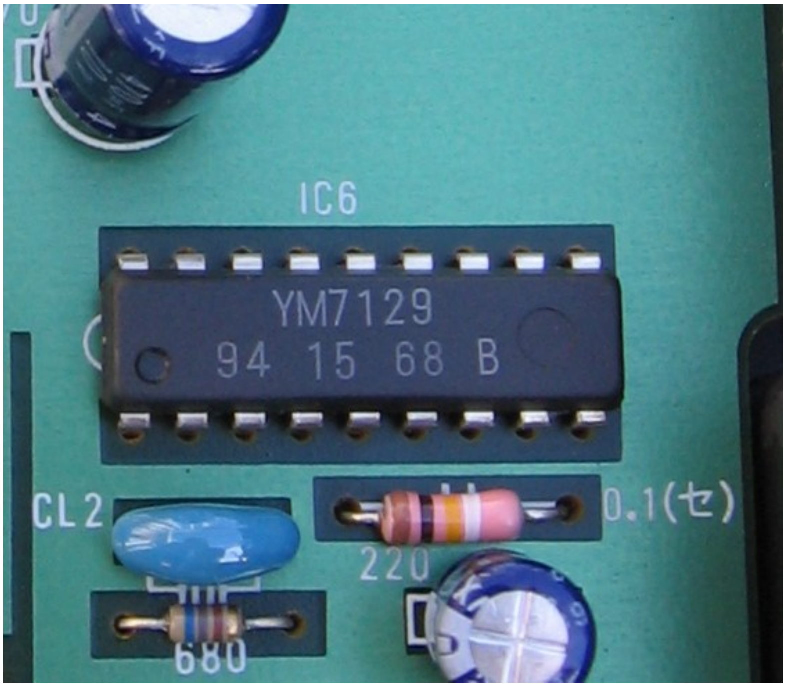
The New Kid on the Block : OPK2

New to version 1.105 is the OPK2!

A lesser known chip, the OPK2 was used in Yamaha PSS-280 and PSS-380 home keyboards. It has 8 channels of 2 operator FM and 4 channels for drum samples played from an internal ROM.

Unique among Yamaha's smaller chips, the FM channels allow each operator to have a truly independent pitch, which is great for detuning effects and for bells and xylophones.

Another nice effect is the software-based tremolo, where the CPU modulates operator volumes, which makes tremolo much more flexible than on chips where the effect is done in hardware.



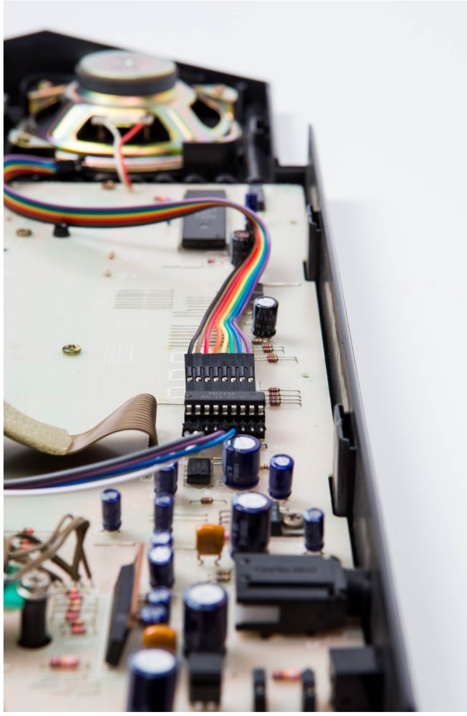
A nice sound chip looking very inconspicuous...

The drum channel play six different classic samples:

- Bassdrum (with a good amount of girth)
- Snare (a soft low thudding snare suitable for all styles)
- Closed hihat (sharp and percussive)
- Open hihat (with a great sizzle)
- Tom (has a nice low end)
- Ride cymbal (great for Jazz)

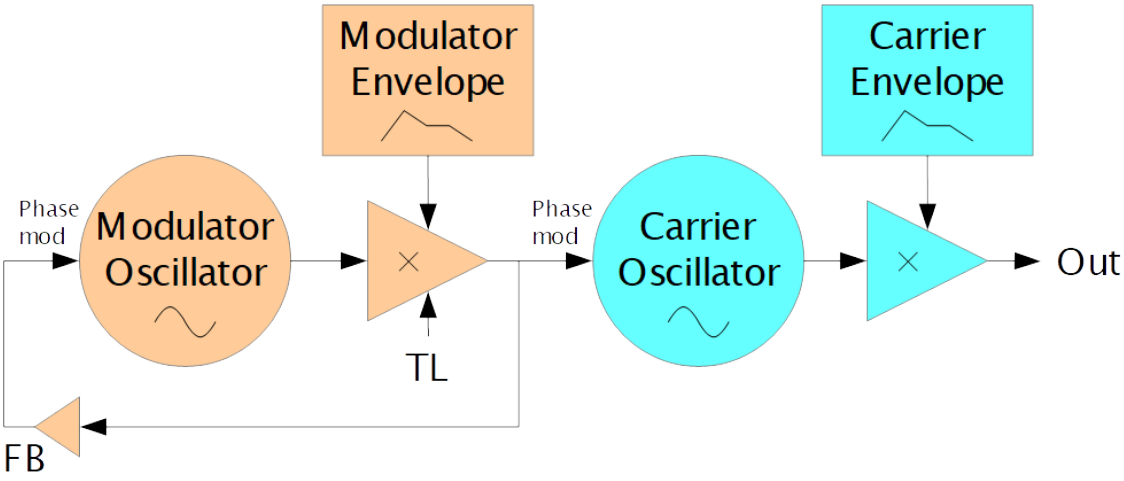
Thanks to our advanced capture techniques, we were able to analyze and recreate this chip without needing any documentation or datasheets, letting people discover a previously mysterious and forgotten FM synthesizer.

(Note: the OPK2 drum samples were recreated using modal synthesis to match the copyrighted originals as close as possible without infringement)



In the belly of the beast.

How it Works



The OPLL, OPL2 and OPK2 chip series have two oscillators per voice:

- The carrier controls the volume. It generates a pure sine wave base for the sound.
- The modulator controls the brightness. It distorts the carrier to build every other harmonic.

This oscillator-controlled distortion process uses phase modulation. For instance, if the modulator wave is at its peak and the modulator's volume is set at 47, the carrier's phase will be offset by 360°. This extremely fast changing of the waveform's phase offset messes it up, turning a simple pure sine into a bright and complex wave, and this very fast phase change effectively modulates the carrier's frequency really fast (hence the name “Frequency Modulation” synthesis).

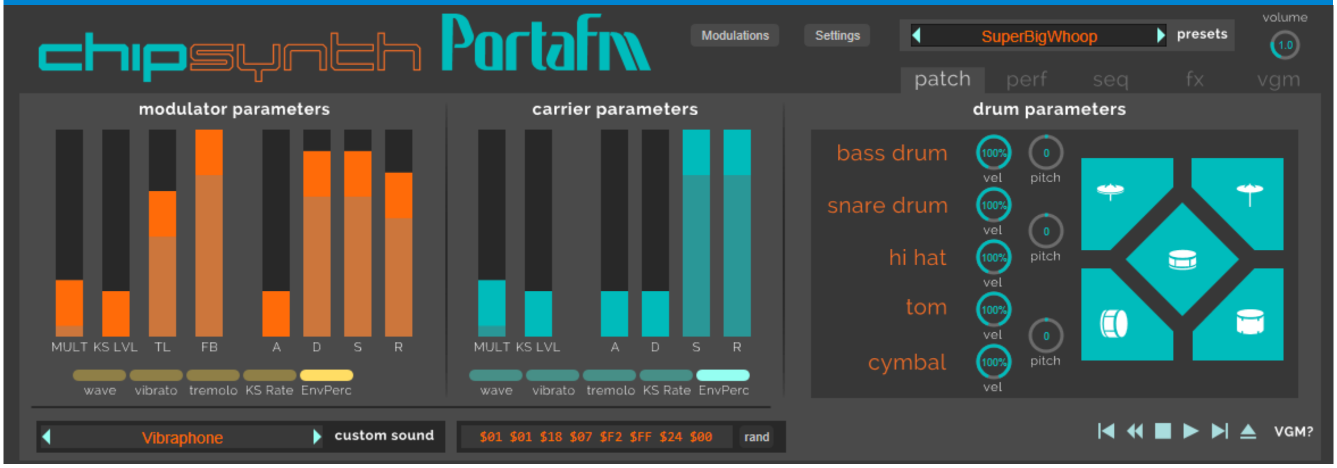
This process makes it very easy to control the brightness: all you have to do is to change the modulator's volume to go from the purity of the perfect sine wave into a hot buzzy distorted grindy mess – and the OPLL/2/K2 chips have a dedicated envelope especially for that! By changing the modulator envelope settings, you can easily build tones that start off bright and then become darker over time, which is an essential feature of a lot of very popular instruments such as guitars and xylophones. This also makes it easy to produce the short but flabby attack of a brass instrument by giving it just a bit of attack.

As if that weren't enough, the modulator can modulate not only the carrier, but also itself. This changes the sound of the modulation by filling in the gaps in the spectrum, which produces the bright saw-wave-like timbre that we all love. If pushed too far, it will even start to go into chaos, the modulator's wave devolving into a cascade of white noise, perfect for drums and sound effects.

All OPLL/2 chips have 9 channels, which means that it contains 18 sine oscillators in total. The OPK2 is similar but has 8 channels (polyphony will be reduced to 8 voices if OPK2 is in use). Plogue PortaFM lets you layer two different chips simultaneously, which lets you build thick, rich layerings.

In addition to this, you can sacrifice 3 channels to add the possibility of generating a set of 5 rhythm tones (bass drum, snare, hihat, tom and cymbal). In PortaFM, this is done on a separate third OPLL/OPL2 emulation core, which means that you retain your full 9 voice polyphony.

FM Patch



Presets : Loads and saves PortaFM presets (patch + perf + seq + modulations + settings + fx)

Volume : Final output volume.

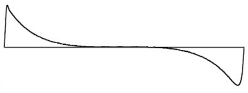
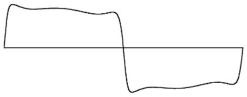
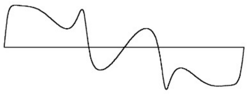
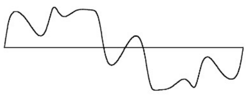
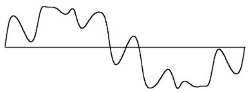
Modulator and Carrier

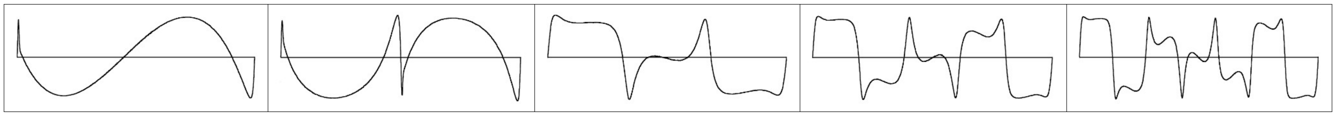
Frequency Multiplier (Mul) : Multiplies the oscillator frequency by 0.5, 1 to 10, 12 or 15.

If you multiply both the modulator and carrier frequency by the same amount, you will shift the pitch of the whole sound. For instance, if you have a sound where the multipliers are 5 and 1 and change this for 10 and 2, the sound will go up by one octave.

The ratio between the frequency of the modulator and carrier controls the “waveform” of the sound:

Modulator	Carrier	Sound Equivalent	Modulator	Carrier	Sound Equivalent
1	1	Saw wave	1	2 or more	Bandpassed saw wave
2 or 4	1	Square or triangle wave	2 or 4	3, 5, 7, 9...	Bandpassed square wave
3	1	Pulse wave	3	2, 4, 5, 7...	Bandpassed pulse wave
5, 7, 9+...	1	Ring modulation	5, 7, 9+...	2, 3, 4, 6...	Bandpassed ring modulation

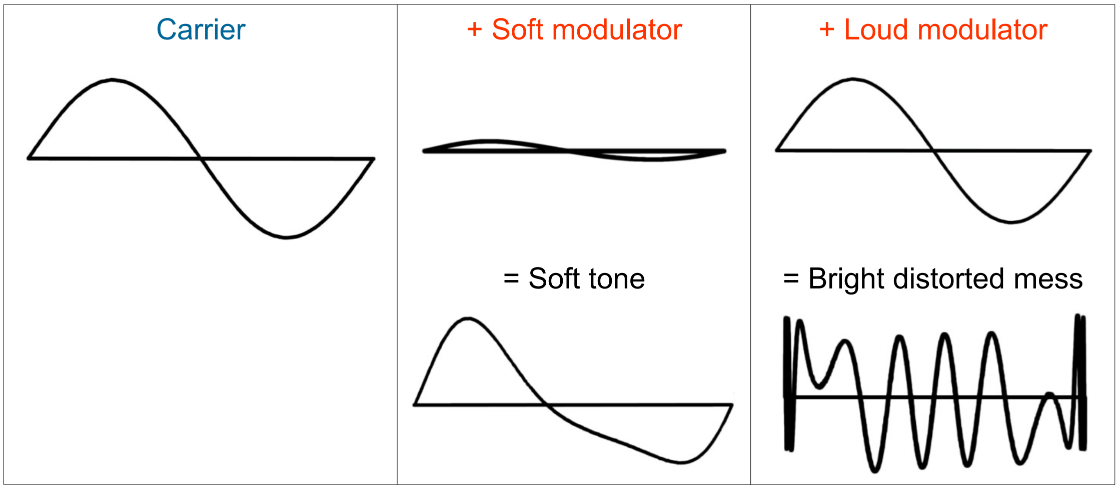
1:1	2:1	3:1	5:1	7:1
				
1:2	2:3	3:2	5:3	7:4



Keyboard Scaling to Level (KSL, KS LVL) : Volume reduction in higher octaves. Key scaling starts at G2 (2½ octaves below middle C).

KS LVL	0	1	2	3
Half volume (TL -8)	Off	Every 4 octaves	Every 2 octaves	Every octave

Amplitude (TL) (Modulator only) **:** Amplitude of modulator. By changing the volume of the modulator, you change the brightness of the sound : the louder you make the modulator, the more the carrier gets distorted and the more harmonics you add to the sound. The amount of modulation is multiplied by 2 every time this is increased by 8 values.



Special Case : When using the OPL2 chip and enabling the additive algorithm, both the Modulator and the Carrier have a TL setting, which is used to set their volume. In additive mode, Modulator TL will still also affect the feedback depth (which will also vary with velocity and layer volume).

Feedback (FB) (Modulator only) **:** How much the modulator will modulate itself. This lets you change the spectrum of the modulation: at 0, the spectrum will vary quite a lot from harmonic to harmonic, especially if the modulator level is high. At 7, the self-modulation will fill in the volume of harmonics, removing gaps, so that the spectrum is a lot more monotonic, making the result brighter and more like a saw wave.

The level of self-modulation is also affected by the modulator volume (TL). The effect will be to shear the modulator waveform. At very high values, the oscillator will start to self oscillate at very high pitch (16572hz), so the following limits on TL should be respected unless the intended result is noise:

FB level	Max modulator level (TL) before self-oscillation
7	39

6	47
5	55
4 and less	No limit (can't self oscillate)

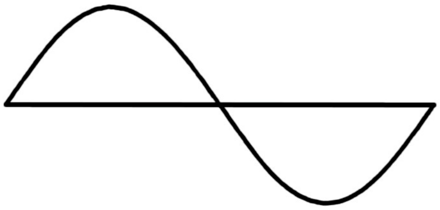
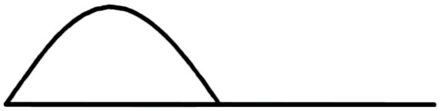
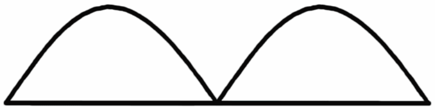
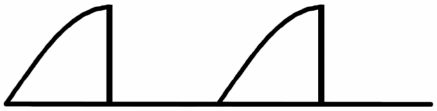
Attack (A) : Attack rate. Note : if you set this to 15 (maximum), the attack will be infinitely long, so you won't get any sound!

Decay (D) : Decay rate.

Sustain (S) : Sustain level.

Release (R) : Release rate. This rate is overridden in some conditions.

Wave : Selects a half-sine or other waveform instead of the default sine wave:

Wave				Uses
OPLL/OPK2	OPL	OPL2	Graph	
SINE	SINE	SINE		Soft sounds, flutes, mallets, bright saw-wave sounds such as brass
HALF	N/A	HALF Half-sine		Distorted sounds, guitars, harpsichords, clavinets, accordions, saxophones
N/A	N/A	DUAL Abs-sine		Distorted sounds, guitars, harpsichords, clavinets, accordions, saxophones. Increases pitch by 1 octave.
N/A	N/A	QURT Pulse-sine		Even more distorted sounds, electric guitar. Increases pitch by 1 octave.

Vibrato : Allows the built-in vibrato to be applied. Rate: 6.07hz, Depth: ±13 cents.

Tremolo : Allows the built-in tremolo to be applied. Rate: 3.7hz, Depth: -5dB.

Keyboard Scaling to Rate (KS Rate, “KSR”) : Speeds up the envelope in the higher notes.

KS Rate	Envelope speed up
On	2x faster every octave
Off	2x faster every 8 octaves

Envelope Percussive Mode (EnvPerc) : Turns off the holding of the envelope when reaching the sustain level – the envelope will automatically go to release.

Additive Mode (OPL2 Modulator only) **:** This disables the frequency modulation and simply adds the modulator and the carrier together, which can be useful for sounds like organs. This enables the Carrier TL setting.

Custom sound preset loader : This lets you load a classic OPLL/OPL2 patch as a custom sound, which is a great starting point for designing a patch. PortaFM has the full complement of presets that are contained in the instrument ROMs of every OPLL and its variants, plus a full GeneralMIDI bank from Sneakernets. This only affects the custom sound part of the sound (essentially the settings on the patch tab).

Randomize patch : Replaces custom sound settings with random ones.

Coarse Tune (OPK2 only) **:** Change the pitch of the Modulator or the Carrier independently in semitones. This allows for very flexible tuning and is useful for sounds like bells, xylophones and so forth.

Fine Tune (OPK2 only) **:** Change the pitch of the Modulator or the Carrier independently in cents. This lets you detune the Modulator from the Carrier, which allows for patches with tone that varies through time as the phase difference drifts, especially of the HALF waveform is used on the Carrier.

Fixed Tune (OPK2 only) **:** Forces the operator to always keep the same pitch (based on middle-C), removing the influence of the keyboard. This can be used for effects such as making the Carrier pitch a very low fixed frequency, making it act as a LFO of sorts, giving a more square-wave quality to timbres.

Software Tremolo Rate (OPK2 only) **:** Change the rate of the tremolo.

Software Tremolo Level (OPK2 only) **:** Change the depth of the tremolo.

Equivalent Settings to Analog Synthesizers

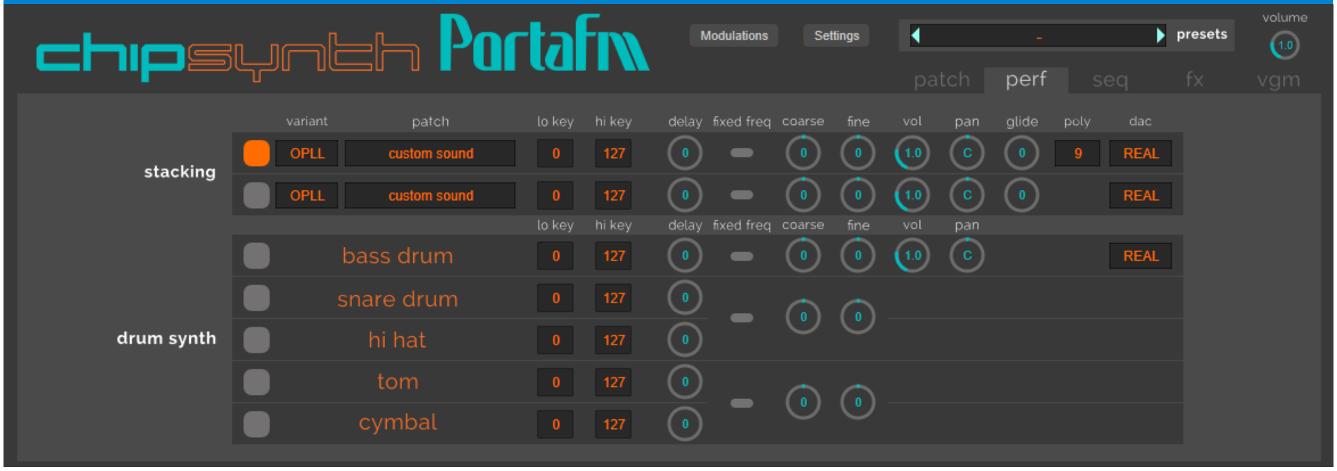
Analog	OPLL/2 Osc	OPLL/2 Setting	Comments
Waveform	Modulator	Frequency Multiplier (Mult)	Low values
Ring modulator	Modulator	Frequency Multiplier (Mult)	High values
Filter cutoff	Modulator	Amplitude (TL)	Max this + FB = noise
Filter resonance	Modulator	Feedback (FB)	Inverse
Filter keyboard track	Modulator	Keyboard scaling to level (KS LVL)	
Filter env A D S R	Modulator	A D S R (attack, decay, sustain, release)	
Filter env keytrack	Modulator	Keyboard scaling to rate (KS Rate)	
Filter LFO	Modulator	Tremolo	
Filter bandpass mode	Carrier	Frequency Multiplier (Mult)	
Vca env A D S R	Carrier	A D S R (attack, decay, sustain, release)	
Vca env key track	Carrier	Keyboard scaling to rate (KS Rate)	
Vca volume key track	Carrier	Keyboard scaling to level (KS LVL)	
Vca LFO	Carrier	Tremolo	
Oscillator octave	Mod. & Car.	Frequency Multiplier (Mult)	Multiply both by ½, 2, 4...
Oscillator LFO	Mod. & Car.	Vibrato	Diff. Values = phasing
Distortion	Mod. & Car.	Waveform	

Drum pad section

Velocity : Individual volume of this drum sound.

Pitch : Pitch offset on this drum. Snare and hihat share the same pitch, as do Tom and cymbal. The hihat and cymbal are generated from a combination of both the pitches.

Performance (Perf)



Melodic Section

On/off : Turns layer on/off.

Variant : Which OPLL/2/K2 variant is used:

- OPLL chips are much grittier, with lots of envelope stepping noise, lower bit depth and less precise pitch. Each variant has different built-in patches. They were used in cheaper home keyboards and on the MSX computers.
- The OPL2 chip is much cleaner and more precise. It can use additive mode and the abs-sine and pulse-sine waveforms. It was famously used on the IBM PC on AdLib sound cards (where it became the de-facto standard), and also in arcade machines and slightly fancier home keyboards. This setting is also be used for OPL sounds, since the OPL is identical to the OPL2 except for lacking the ability to select waveforms other than sine.
- The OPK2 chip is similar to the OPL2 and also very clean and precise. It allows for true independent tuning of the Modulator and Carrier, making it well suited for inharmonic sounds. This also enables the software tremolo, which allows for a variable rate and depth. Note that it has a maximum polyphony of 8 voices (polyphony will be automatically reduced if OPK2 is in use).

Patch (OPLL variants only) : Selects between the user patch (the one from the patch tab) or a built-in patch from the OPLL chip. OPL2 and OPK2 don't have this setting and can only use the custom patch.

Lo key, hi key : Limits the range on which this layer plays.

Delay : Delays note-on.

Fixed freq : Removes the effect of the keyboard on pitch. MIDI note 36 (C2) is played instead.

Coarse tune : Transposes the pitch (in semitones).

Fine tune : Detunes the pitch (in cents).

Volume : Overall amplitude of this layer.

Pan : Panning of this layer.

Glide : How much time the pitch takes to glide to new notes. Set this to 0 to turn off gliding.

Polyphony : Switches between polyphonic mode (2-9 voices) and monophonic voices (1 voice).

Re-Trig (monophonic only) : Retrigger the voice when playing a legato.

Re-Vel (monophonic only) : Change the velocity when playing a legato. (more expressive but clicky)

DAC : Allows the usage of a more accurate DAC than the original chip.

Setting	Result
REAL	Original 9-bit DAC measured from analog output
9-bit	“Perfect” 9-bit DAC (all steps are equal)
12-bit	“Perfect” 12-bit DAC, removes most digital noise

OPL2 has different DAC modes:

Setting	Result
REAL	Original 10 bit DAC with 8 levels of companding.
13:2	13 bit DAC with 4 levels of companding.
OPL3	“Perfect” 16-bit DAC identical to the OPL3.

Drum Section

On/off : Turns layer on/off.

Variant : Which OPLL/2/K2 variant is used for drum mode:

- OPLL: crunchy, low resolution drums. OPLL drum settings can’ be edited (the chip uses a ROM).
- OPL2: smoother, more precise. OPL2 drum settings are editable.
- OPK2: sampled drums played from an on-chip ROM (non-editable), has closed & open hihat.

Edit (OPL2 only) : Change the settings for drums. The drums work as follow:

- Bass drum: Same as the regular melodic patch.
- Snare: The snare drum is a mix of noise and square wave tuned by the hihat’s frequency.
- Hihat: This drum uses a pulse pattern combining the hihat and cymbal oscillators and noise.
- Tom: A simple decaying sine wave!
- Cymbal: This is a complex pulse pattern based on the hihat and the cymbal oscillator.

For the snare, hihat, toms and cymbal:

- **KSL (Key scale to level), A (Attack), D (Decay), S (Sustain), R (Release), Tremolo, KS Rate (Key scale to rate), EnvPerc (Envelope percussive mode) :** These settings act on the drum's volume and are identical to the matching Carrier settings.
- **Mul, Vibrato :** These setting change the frequency of the underlying oscillator. The oscillators for the snare, hihat, tom and cymbal are recombined together in non-obvious ways to generate drums, so the effect of this is complex and often unexpected.
- **Wave :** This rectifies the drum's waveform in the same way sine waves are modified in the melodic patch. Due to the way the snare, hihat and cymbal are generated using pulses and noise, this might simply cause a volume drop.

Lo key, hi key : Limits the range on which this layer plays.

Delay : Delays note-on.

Fixed freq : Removes the effect of the keyboard on pitch. MIDI note 36 (C2) is played instead.

Coarse & Fine tune : Offsets the pitch. Snare and hihat share the same pitch, as do Tom and cymbal. The hihat and cymbal are generated from a combination of both the pitches.

Volume : Overall amplitude of all the drums. The individual volume of each drum is set on the Patch tab.

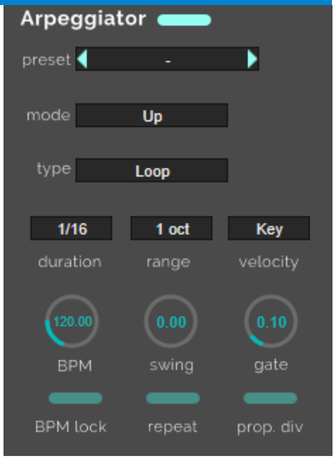
Pan : Panning of the drums.

Open Hihat Key (OPK2 only) : If active, notes above this key will trigger the open hihat rather than the closed one.

DAC : Allows the usage of a more accurate DAC than the original chip. (see above)

Effects (Fx) : Arpeggiator

The arpeggiator lets you turn any incoming MIDI data into arpeggio patterns.



Arpeggiator

preset -

mode Up

type Loop

duration 1/16

range 1 oct

velocity Key

BPM 120.00

swing 0.00

gate 0.10

Preset : Load and save arpeggiation presets.

Mode : Arpeggiation pattern:

Mode	Pattern	[ ] Variant
Up	From low to high	
Down	From high to low	
Up-Down	Low to high to low	Top & bottom notes repeat
Down-Up	High to low to high	Top & bottom notes repeat
Key	Same order as pressed	
Random	Random pattern	Can repeat the same note twice

Type : Switches between continuous arpeggiation (loop) and one-shot

Type	Mechanics	After last note
Loop	Cycles through currently held notes.	Repeat
Single	Notes played together in one close block are arpeggiated together in a single shot.	Stop
Sustain Last		Hold last note

Duration : Time between arpeggiated notes.

Range : Allows automatically adding more arpeggiated notes one octave or two octaves above.

Velocity : Switches between the following velocity modes:

Mode	Velocity
Key	Same as each individual key press
1st	First velocity used for the group of notes
Norm	Always 100

BPM: Tempo at which notes are arpeggiated. Can kept in sync to host BPM with the BPM lock param.

Swing : Allows for swinging between slightly longer notes and slightly shorter notes.

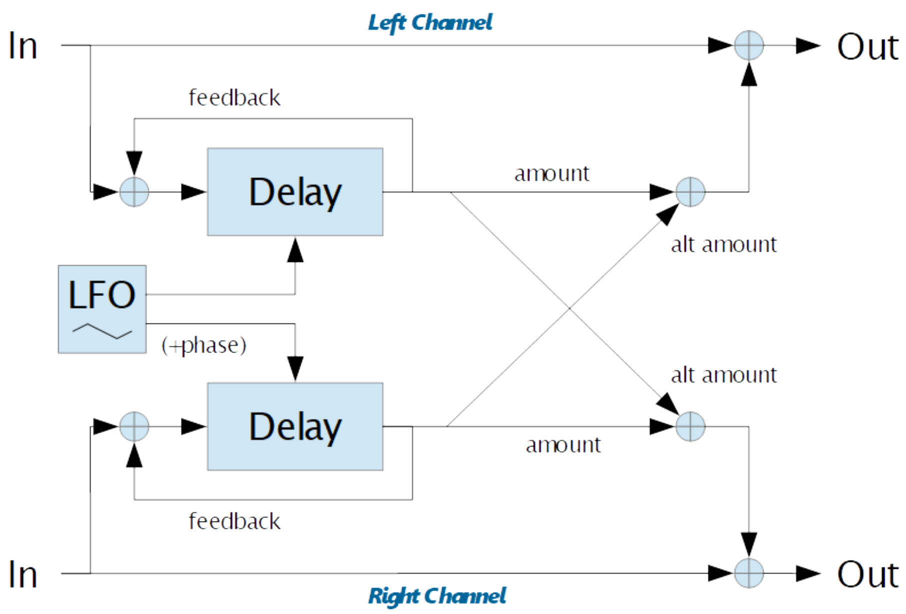
Gate : Shortens notes, inserting more space in between.

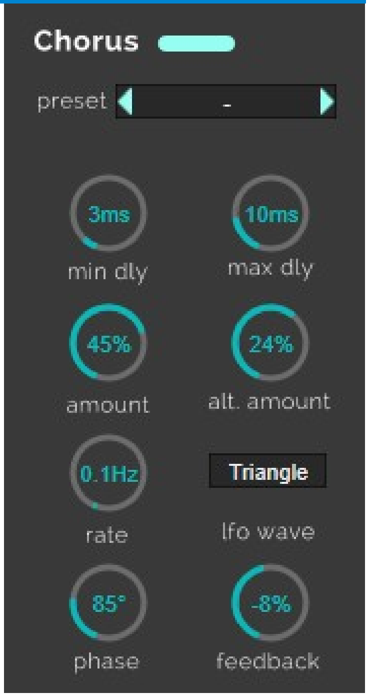
BPM Lock : Automatically forces the BPM to follow host BPM.

One Note Repeat : Allows repeating arpeggiation to be used even when only one single note is played.

Proportional Division : Accelerates repeat rate as more notes are added, keeping total length.

Effects (Fx) : Chorus





preset -

min dly 3ms

max dly 10ms

amount 45%

alt. amount 24%

rate 0.1Hz

lfo wave Triangle

phase 85°

feedback -8%

Based on the famous Dimension D, this chorus will let you capture the classic rich and wide sound that will put your FM sounds in the best light!

On/off : Lets you turn the effect on or off.

Preset : Lets you load chorus settings from a preset.

Min Delay : How long the song is delayed when the LFO is at its minimum level.

Max Delay : How long the song is delayed when the LFO is at its maximum level.

Amount : Mix level of chorused sound.

Alternate Amount : Mix level of chorused sound with channels swapped. Setting this the same as amount will make the effect mono, setting this to the inverse will give the maximum stereo field.

Rate : How fast the LFO modulates the delay between minimum and maximum length.

Lfo Wave : Switches the LFO waveform between triangle wave (default) and sine wave.

Phase : Phase offset of the LFO to the second channel.

Feedback : How much signal is recirculated into the chorus.

Effects (Fx) : Ambience Reverb

Ambience is a very smooth and flexible stereo reverb effect created by Magnus Jonsson, known mostly for its Ambience VST/AU version.

In particular, the decay time can be shaped to different values in the low/mid/high frequency ranges.

On/off : Lets you turn the effect on or off.

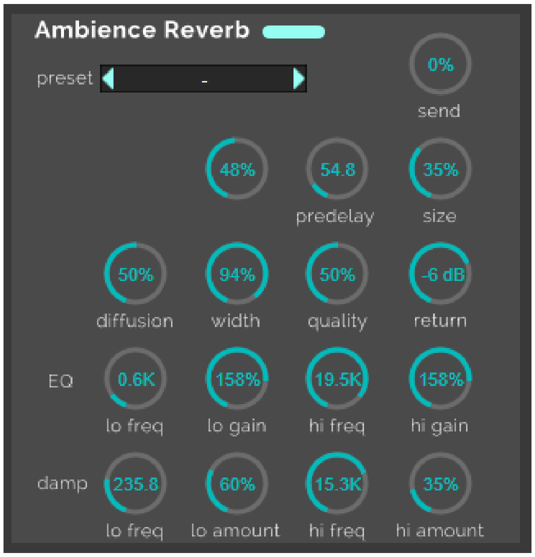
Send : Controls the level of reverb.

Preset : Lets you load reverb settings from a preset.

Decay : How long the reverb decays.

Diffusion : How smooth the decay tail is.

Size : Time between the echos (simulates a small or large room).



Ambience Reverb

preset -

send 0%

48%

predelay 54.8

size 35%

diffusion 50%

width 94%

quality 50%

return -6 dB

EQ lo freq 0.6K

EQ lo gain 158%

EQ hi freq 19.5K

EQ hi gain 158%

damp lo freq 235.8

damp lo amount 60%

damp hi freq 15.3K

damp hi amount 35%

Predelay : Amount of delay between the original sound and its reverb tail.

Width : Amount of stereo separation (how different the reverb is on the left vs right side).

Quality : Increases the complexity of the reverb tail, at the cost of requiring more CPU to process.

Output : Volume of reverb.

Equalizer Lo Freq : Frequency under which the low gain is applied.

Equalizer Lo Gain : Gain applied on low frequencies.

Equalizer Hi Freq : Frequency over which the high gain is applied.

Equalizer Hi Gain : Gain applied on high frequencies.

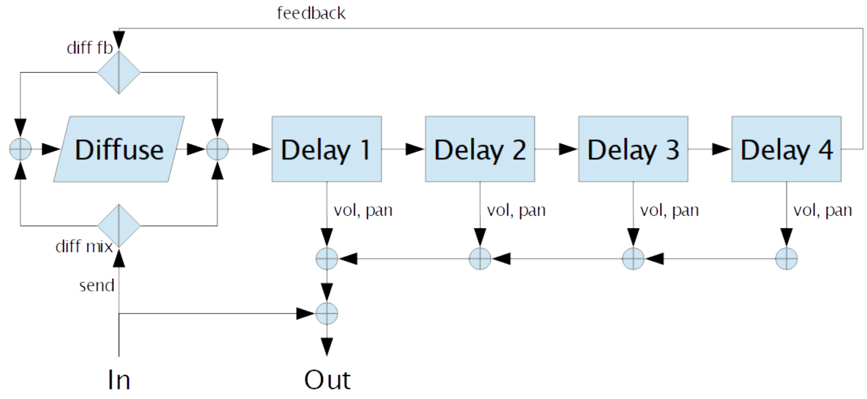
Damping Lo Freq : Frequency under which low damping is applied.

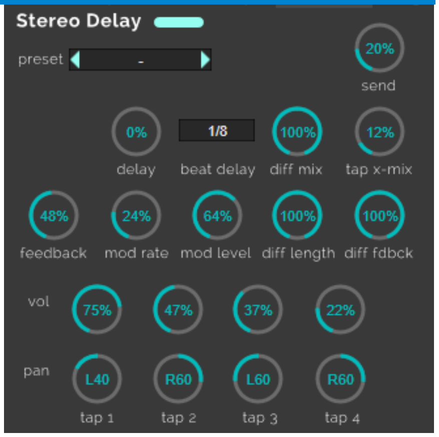
Damping Lo Amt : Decay time change under low damping frequency.

Damping Hi Freq : Frequency over which high damping is applied.

Damping Hi Amt : Decay time change over high damping frequency.

Effects (Fx) : Stereo Delay




Stereo Delay

preset -

send 20%

delay 0%

beat delay 1/8

diff mix 100%

tap x-mix 12%

feedback 48%

mod rate 24%

mod level 64%

diff length 100%

diff fdbck 100%

	tap 1	tap 2	tap 3	tap 4
vol	75%	47%	37%	22%
pan	L40	R60	L60	R60

The stereo delay is custom made for chipsynth and creates a repeating pattern of four delays. Along with extensive modulation and diffusion capacities, this allows for huge endless spatial sounds.

On/off : Lets you turn the effect on or off.

Send : Controls the level of the delay effect.

Preset : Lets you load delay settings from a preset.

Delay : Time between repeats (in non-musical time).

Beat Delay : Time between repeats in beats. This is added to the delay (see above).

Diffusion Mix : Mix balance between crisp reflections and diffuse reflections.

Tap X-Mix : Short-circuits the normal order of reflections.

Feedback : How much signal is reinjected back after the 4th reflection.

Modulation Rate : How fast the delay time is modulated.

Modulation Level : How much the delay time is changed by the modulation.

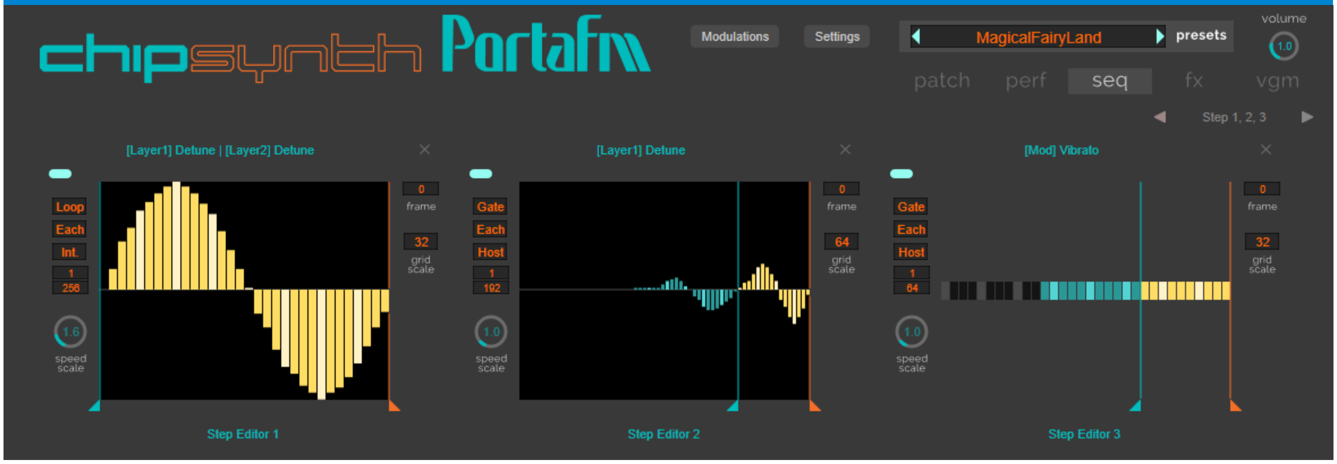
Diffusion Length : Length over which diffuse reflections get spread.

Diffusion Feedback : How much of the feedback gets diffused each time the signal loops around.

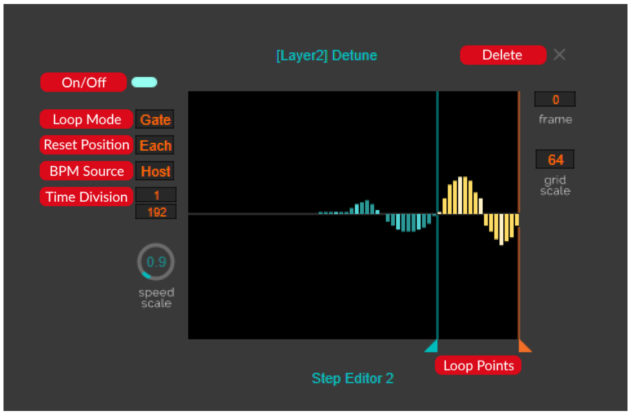
Tap 1..4 Volume : Amplitude of each reflection.

Tap 1..4 Pan : Panning of each reflection.

Step sequencers (Seq), Modulation and Settings



Sequences



Loop Mode : Switches sequence between one shot, loop with intro and note-off, or loop only

Reset Position : Key sync type

BPM Source : Tempo or note effect on step rate

Time Division : Musical steps duration. Ex: Setting this to 1:16 makes each step last a sixteenth note.

Speed Scale : Multiplies step rate.

Frame : Which one of the subsequences is used. This can be modulated.

Grid Scale : Sets the number of steps for the sequence.

Loop Points : Start and end of the loop. (must be in 'gate' or 'loop' mode)

Display : Lets you draw the shape of your modulator.

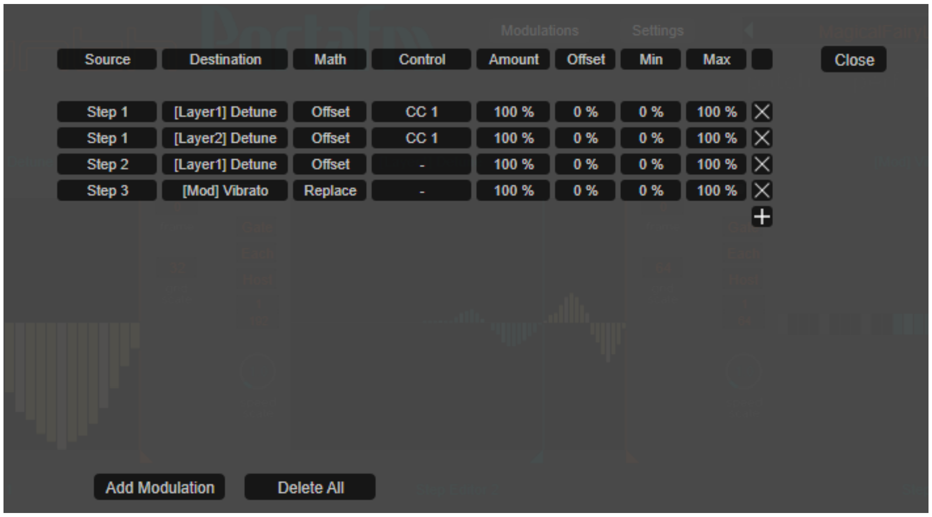
Loop Mode	Typical use	Effect	Note Off
One Shot	Envelope	No loop points. Plays the full sequence, then stops.	Ignored
Note Gate	Sustained envelope + LFO	Starts at beginning and loops between loop start and end. Jumps to points after loop end on note off.	Jump to after loop
Loop Always	LFO	Loops forever from loop start to loop end. All points before loop start and end are ignored.	Ignored

Reset Position	Type	Synchronization
Each Note	Polyphonic*	Resets on every note
First Note	Monophonic	Resets on the first note of a group
Last Note		Resets on every note
Bar Pos		Musically synchronized with the start of the bar

* Note : The OPLL only supports one custom sound at the same time. Because of this, all patch settings cannot be modulated polyphonically, and instead behave as if the “Last Note” mode has been selected.

BPM Source	Step rate
Host BPM	Follows song BPM.
Internal BPM	Always 125 BPM. Setting time division to 1:96 will yield 50 steps per second.
Note Period	Step duration is scaled to note. 1:128 gives 1 step per period.

Modulation



Source	Destination	Math	Control	Amount	Offset	Min	Max
Step 1	[Layer1] Detune	Offset	CC 1	100 %	0 %	0 %	100 %
Step 1	[Layer2] Detune	Offset	CC 1	100 %	0 %	0 %	100 %
Step 2	[Layer1] Detune	Offset	-	100 %	0 %	0 %	100 %
Step 3	[Mod] Vibrato	Replace	-	100 %	0 %	0 %	100 %

Source : Step sequencer or MIDI value which modulates the parameter.

Destination : Selects which parameter gets modulated. Not all parameters are available for every modulator.

Math : How the modulation is combined with other value sources. Note: MULTIPLE MODULATION TYPES CANNOT BE USED ON A SINGLE DESTINATION. (they must all be Offset or all be Scale)

Math	Effect
Offset	All sources are added together.
Replace	The “base” value is ignored and the modulation takes over. DESTINATIONS MODULATED USING THIS CAN ONLY HAVE ONE SINGLE MODULATOR.
Scale	All sources are multiplied together.

Control : Adds second modulation source to scale the amount of modulation.

Amount : Scales the amount of modulation to the parameter.

Offset : Offsets modulator values before they are scaled and applied.

Min : Limits modulator input values to a minimum value.

Max : Limits modulator input values to a maximum value.

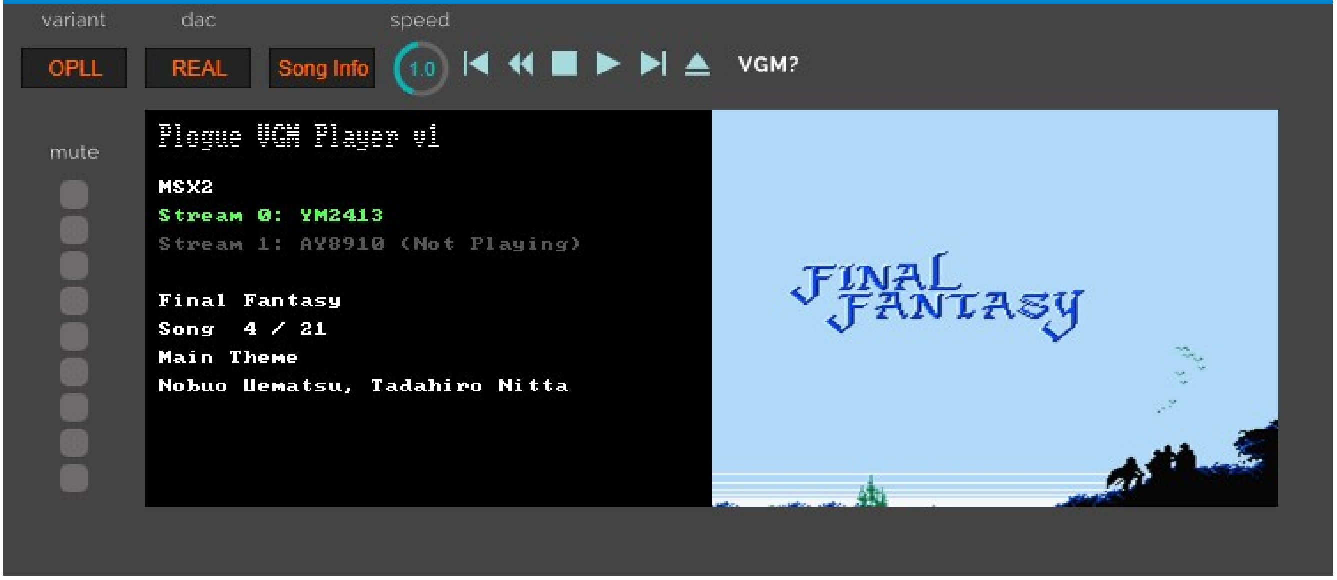
Settings

Pitch Bend Range : Range of pitch bend wheel (in semitones).

Velocity Tracking (Veltrack) : Amount of velocity effect on note volume.

Tuning (cents) : Master detune. Allows you to shift the pitch of everything, for instance when not using A=440hz tuning.

VGM *(Video Game Music) : Vintage game soundtrack playback*



```
Plogue VGM Player v1

MSX2
Stream 0: YM2413
Stream 1: AY8910 (Not Playing)

Final Fantasy
Song  4 / 21
Main Theme
Nobuo Uematsu, Tadahiro Nitta
```

VGM files contain raw register data lifted from vintage games which are meant to be directly interpreted by their target sound chip. Think of it as a very crude version of MIDI which also contains patch data and parameter automation.

Early on in the chipsynth series design it was decided that every member of the family should be able to play its of its legacy formats. This is mostly used as a regression test bed of our emulator cores, and also because we love those soundtracks so much and we feel they could inspire as well.

variant : Which OPLL variant is used. Different variants have different built-in patches, which makes for interesting experiments. Note OPLL-2 is added for completeness and study, its pitch registers are mapped DIFFERENTLY, so normal OPLL VGM streams WILL sound wrong if you use it. OPL/OPL2/OPL3 vgm files automatically use OPL/OPL2/OPL3 mode and ignore this setting.

dac : Controls the level of the delay effect.

info : See the M3U or the Registers (more features in the future)

speed : change the data playback rate (cannot easily be tweaked to clear BPM for technical reasons)

mute : Mute any of the 9 tone channels. Good to understand how things were made, and for VGM study.

Note 1: Only VGM streams which contain data for OPLL (YM2413 and its variants) and OPL1/2/3 chips will play. For compatibility, AY8910 (MSX PSG) and SN76489 (Master System PSG) are supported in VGM playback only. Y8950 (MSX-AUDIO) will play FM but no ADPCM . Only the 9 channels from OPLL/2 can be muted/grabbed/set to solo. The rest of channel streams are ignored.

Using chipsynth PortaFM as a Standalone App

PortaFM can be launched by itself and played live via MIDI keyboard or other MIDI controller. The standalone version of PortaFM effectively makes your computer, audio hardware and MIDI keyboard into a virtual synthesizer that can be played independently of other programs. Unlike using it as a plug-in within a sequencer, your recording ability is limited and you can not edit your performance (though you can use various audio software programs for this).

Launching PortaFM in Standalone Mode

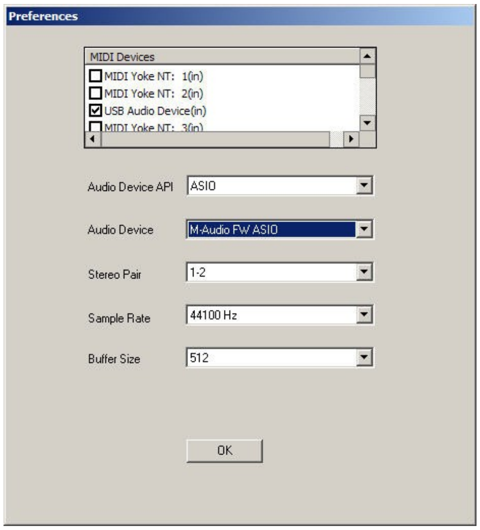
First, make sure that you have followed the instructions in the installation section of this manual. Be certain that your audio/sound interface and MIDI hardware interfaces are properly connected to the computer, your speakers or headphones are connected and everything is powered up.

To launch PortaFM as a standalone application, click on the PortaFM logo on your desktop or go to the Program Files or Applications folder and launch “PortaFM”.

Basic Setup Information for Standalone Mode

To use the standalone version, you have to configure the Audio and MIDI settings in the Preferences dialog box (found in the Tools menu) before you can play. When used as a plug-in, the host sequencer or tracker program has already set up its audio and MIDI connections, and the PortaFM “plugs in” to them. However, with standalone operation PortaFM communicates directly with your audio and MIDI interface. Setup for Mac and Windows computers is similar, except where indicated. Note that if you change your audio interface, you will almost certainly need to readjust these settings.

Call up the Preferences setup dialog from the Tools menu on the PortaFM standalone interface. You’ll see drop-down menus for MIDI Device, Audio Devices, Stereo Pair, Sample Rate and Buffer Size.



Preferences

MIDI Devices

□ MIDI Yoke NT: 1(in)
□ MIDI Yoke NT: 2(in)
☑ USB Audio Device(in)
□ MIDI Yoke NT: 3(in)

Audio Device API: ASIO

Audio Device: M-Audio FW ASIO

Stereo Pair: 1-2

Sample Rate: 44100 Hz

Buffer Size: 512

MIDI Device Menu: All supported & installed MIDI interfaces are available here. Select the desired MIDI device from the list to send and receive MIDI to it.

Audio Device Menu: All supported (and installed) audio interfaces are available in this drop-down list. Select the desired audio device from the list.

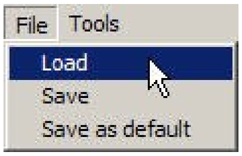
Stereo Pair: Here you can define which of the stereo outputs should be used. Many pro audio devices have a variety of outputs, so you may choose which of these you would like PortaFM to output through.

Sample Rate: Depending on the sound card and driver you are using, various sample rates are available. Set the desired sample rate here.

Buffer Size: The buffer size setting will determine the delay between pressing a key on your MIDI keyboard and hearing the sound (a/k/a ‘latency’). The default buffer size of 512 samples typically works well, but smaller buffer sizes will give a faster response (lower latency) and higher buffer sizes will give better audio performance (more polyphony and higher fidelity). Most modern computers and audio interfaces can handle a buffer size of 512 samples without a significant reduction in polyphony. If the sound is breaking up or crackling when a note sounds, then first check that the audio connections and wiring are good. Then, try a larger audio buffer size setting. Please note that there is typically a trade-off between higher buffer sizes (polyphony and sound fidelity) and lower buffer sizes (faster response or lower latency). Also note that the sound card buffer size settings determine latency, rather than PortaFM Player itself.

Once you have your Audio and MIDI set up, and have loaded one of the snapshots, you can begin playing PortaFM. Try playing a key on your MIDI keyboard. If the MIDI and audio configurations are correct, you should hear the corresponding synth note. If not, check the MIDI connections and wiring, and the MIDI output channel of your MIDI keyboard. Also check that the channel is specified correctly. If you are hearing the notes play, then the basic configuration is complete, and you are ready to use PortaFM.

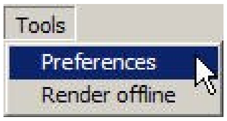
File Menu for Loading and Saving Snapshots in the Standalone



Configuration presets (.fermatax files) for PortaFM can be saved and loaded. This gives the user the ability to customize instruments setups to suit personal preferences and save configurations for convenient future use. The File menu choices are:

- *Load*—any saved configuration preset files in Fermata format can be loaded by clicking on this choice and selecting the desired file.
- *Save*—any configuration can be saved by clicking on this choice, typing a name for the custom preset and saving to a desired location.
- *Save as default*—any settings can be saved as part of the default, to be loaded automatically at the time the PortaFM is booted in standalone mode.

Tools Menu in Standalone ONLY

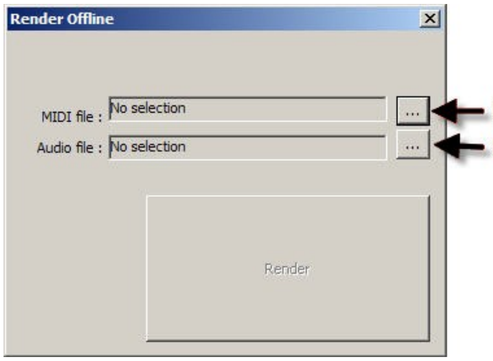


The Tools menu includes:

- *Preferences*—as described in the basic setup information above.
- *Render offline*—MIDI files can be rendered to audio offline using this feature.

To use the Render Offline feature:

1. Click on the Load MIDI file button.
2. Select the desired MIDI file
3. Click on the Name audio file button.
4. Name the audio file and specify its location.
5. Once the files are in place, there is just one more step: Click on the Render button, which will render the files to the selected audio file.


Render Offline

MIDI file : No selection

Audio file : No selection

Load MIDI file

Name audio file

Using chipsynth PortaFM as a Plug-In

When used as a plug-in, PortaFM is not a standalone program, but rather a virtual instrument module that seamlessly integrates into your favorite music software program, sequencer or tracker (the “host”, which is modular in order to accept plug-ins).

Using PortaFM as a plug-in has various advantages:

- ♦ MIDI recording and sequencing
- ♦ Easy automation of PortaFM parameters through the use of MIDI CC's or the host's automation.
- ♦ Audio mixing with other plug-ins and effect processing of PortaFM within a single program.
- ♦ Integration with other instruments into a “virtual studio”.

A great thing about plug-ins is that they work with a large variety of compatible music programs: PortaFM can be used as a VST plug-in in many VST music programs, sequencers, supported tracker programs and hosts. It can also be used as an Audio Unit or AAX plug-in.

Plug-in Standard	Description	Windows 64bit	Mac 64bit
VST	VST (Virtual Studio Technology) was developed by Steinberg for Cubase, and is also used by Ableton Live, FL Studio, Studio One, Reason, Bidule, etc.	X	X
Audio Units	The Audio Units (AU) plug-in standard was developed by Apple for Core Audio in Mac OS X. Audio Units is the preferred plug-in format on Mac, used by Apple GarageBand & Logic and MOTU Digital Performer.		X
AAX	AAX plug-ins are designed for Digidesign Pro Tools, used extensively in the pro audio and post production communities.	X	X

Plug-in Use

To use PortaFM as a plug-in instrument, you simply launch your host music application/sequencer first and then launch chipsynth PortaFM from within it. Make sure that your sequencing host program is properly installed and configured, and that it is producing sound properly. Used as a plug-in, PortaFM’s audio and MIDI data is managed by the host music software.

Each music software application has its own approach to handling plug-in instruments. They each have a different method of installation as well as differing means of loading and accessing plug-ins. It is important to make sure that you refer to the instructions in your music software application’s manual regarding the loading and operation of plug-in instruments.

Windows VST Setup

Make sure the chipsynth PortaFM plugin is within the VST plugin folder used by your host, so that it can be scanned and recognized on startup.

Credits

Plogue chipsynth PortaFM®

Produced by :	David Viens
FERMATA Engine Design/Development :	Plogue Art et Technologie, Inc.
Hardware research and modeling :	David Viens Hubert Lamontagne
Presets :	David Viens Hubert Lamontagne Xenos Soundworks
In-house Testing and QA :	Eric Patenaude
Chorus :	Ross Bencina
Ambience Reverb :	Magnus Jonsson
UX early concepts and logo	JCPP

Acknowledgments

We would like to give special thanks to the following people:

Jonathan Gevaryahu
Matt Gallais (pixoshiru)
Akira Key
Johnny Collazo Jr.
A_Rival
Florian Rehn
Jerome Lebel
XC3N
Kevin Horton (kevtris)
Kevin R. Burke
subz3ro
Peter Swimm
VECTRAX
Dave Miles
Infinite NES Lives
madscient
Mathew Valente (tssf)
Sneakernets
Cuckoo
Alex Hauptmann
Inverse Phase
Andy Lemon
Game Genie Sokolov
Yuzo Koshiro

Thank you!

The information in this document is subject to change without notice and does not represent a commitment on the part of Plogue Art et Technologie, Inc. No part of this publication may be copied, reproduced or otherwise transmitted or recorded, for any purpose, without prior written permission by Plogue Art et Technologie, Inc.