

# 8bit Anthem Documentation

- \*About the Package
- \*About UHM And Our Wavetables
- \*Understanding Each Wavetable
- \*Miscellaneous Information
- \*Download & Update
- \*List of Wavetables



## About the Package

“8bit Anthem” is a HIVE2 expansion featuring original wavetables.

### How to Install Presets

---

- Open Hive
- Switch to the **Preset Browser**
- Right-click on the **User** folder to show a popup menu
- Select the **open in Explorer** (Windows) / **reveal in Finder** (macOS) option.
- Drop the preset folder there.

### How to Install Wavetables

---

- Open Hive
- Open wavetable loading menu
- Select the **open in Explorer** (Windows) / **reveal in Finder** (macOS) option.
- Drop the preset folder there, or any newly created folder inside there, with any name you like.

How does Hive manage wavetables? Here is what Urs says about that :

Currently wavetables are identified solely by their filename and location. If a wavetable isn't found where it was when saving a preset, the engine looks into the preset directory first. Then checks any other directory within the wavetables folder.

(Quoted from a KVR Thread)

So as long as it resides inside the "Wavetables" folder, a folder can be moved or renamed. But you can't **rename** wavetable files.

\*UHM wavetables are not necessary for presets to be loaded. They are just *source codes* for those who want to see the inside.

## How to Install Microtunings files

---

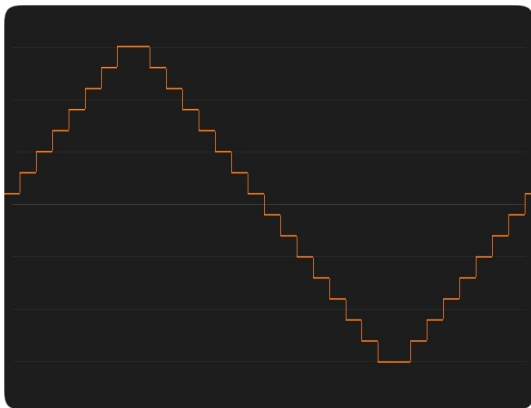
- Open Hive
- Open microtuning files browser
- Right click "User" folder and select the ***open in Explorer*** (Windows) / ***reveal in Finder*** (macOS) option.
- Drop the mtun files there.

# About UHM And Our Wavetables

How do wavetables contribute to 8bit sound?



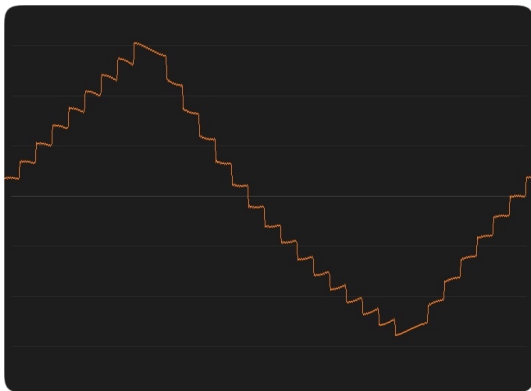
## Pursuit of Real Vibes



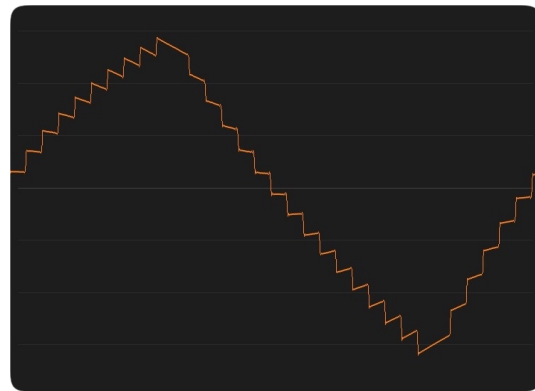
There are many VST instruments that call themselves as "8bit", but not every one has true vibes of classic game consoles.

The image left is an output waveform generated from one of those *8bit-ish* plugin. Though it looks lo-fi, it's not enough to evoke a vivid sense of real 8bit world. It sounds too cold and solid, lacking something that catches your heart. Why?

-- It is because real waveforms from these hardware have more **complex curves**. Looking at real recorded samples, you'll easily notice that no single step is flat. Some have acclivity while others have declivity. Some are concave and others are convex...

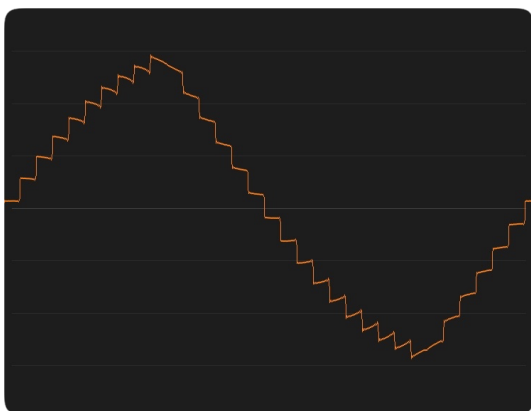


Recorded sample 1



Recorded sample 2

*God is in the details.* All those delicate curves bear **unique harmonics** that are crucial for classic vibes. And what we've done in 8bit Anthem is to completely recreate these details.



Our wavetable

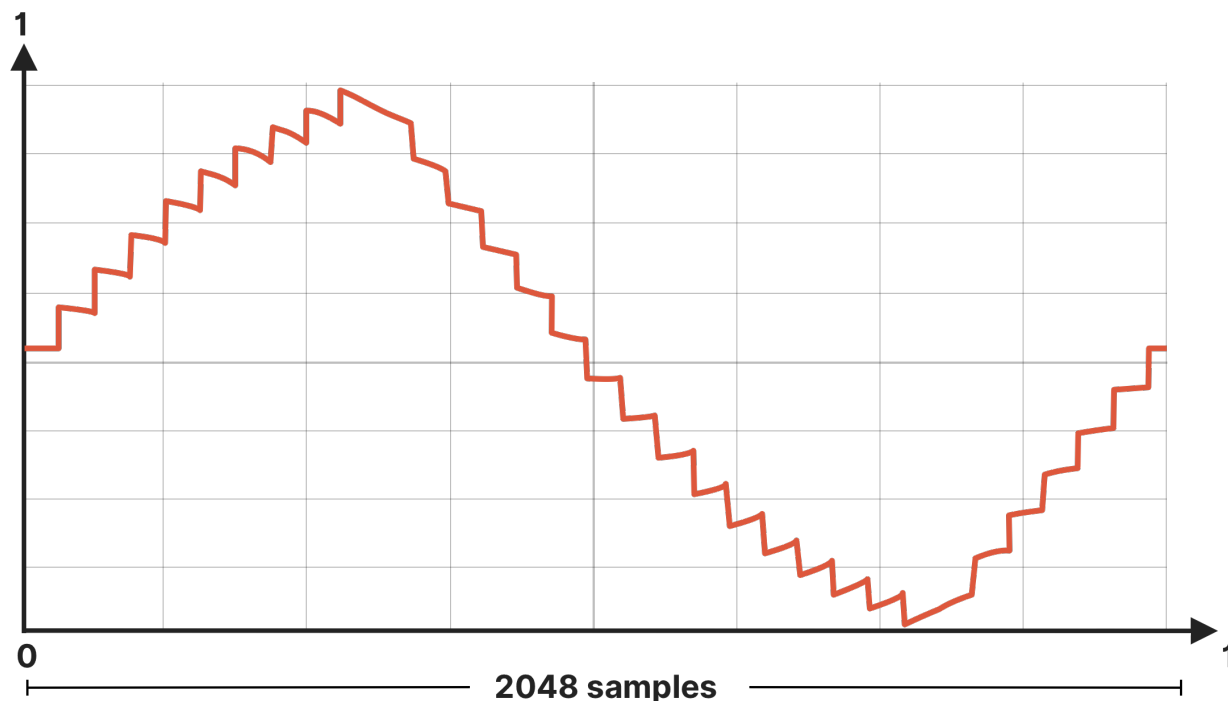
The image left is the output waveform of our wavetable. Using oscilloscope and spectrum analyzer (and of-course, ears!), the characteristics of waveforms are captured as a wavetable.

Written in UHM, every curve is controlled by math, to achieve the best quality sound.

## What is UHM?

UHM(U-He Math) is a scripting language to generate whatever waveforms you want to.

UHM wavetables have 2048 samples resolution and you can draw literally any curves using math and logical operations, which can readily replicate all the finest level characters described above.



UHM files can be edited via any text editors and can be loaded as a wavetable in HIVE, or can be exported as .wav files.

## Compatibility

Though wav files generated from UHM can be also loaded into other wavetable synths than HIVE, the compatibility varies.

For example, **Vital** can load them without any labor, just simply from browser. But **Serum** cannot directly load them from drop down menu. You have to drag & drop them on to Serum and manually convert them.

# Wavetables vs Recorded Samples

Comparing wavetables with recorded samples, both have their pros and cons.

## Sounds

---

Talking of sounds, recorded samples, as a matter of course, excel in its realisticness. They hold every bit of characters of original consoles, including hardware noises, distortion and so on. Even though our wavetables reproduces complex curves of the waveforms, they're **still more hi-fi** in that they don't include any noises, glitches or distortion.

It depends on situations which sound is suitable. Sometimes you may want a cleaner material to which you add your own preferable flavor, or other times you want to insert clean 8bit sounds to a totally hi-fi track...8bit Anthem will be perfect fit in such situations.

8bit Anthem is like a glorified dream—where the warm side is preserved while the dirty side is in oblivion.

## Mobility

---

In terms of editability and mobility, wavetables have great advantages like smaller file size, wider possibility of editing through wavetable synths, or you can re-edit those wavetables through wavetable editors...And as for UHM wavetables, you can even look into the source codes and edit them.

Sample based products and 8bit Anthem have different strong points and therefore can live together.

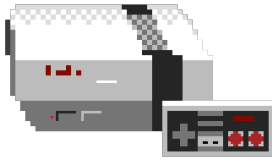
# Understanding Each Wavetable

Taming 1 wavetable  
is worth more than  
Merely having 100 wavetables.





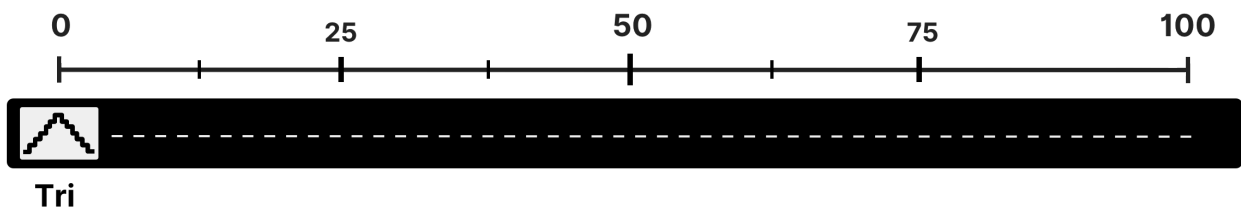
# NESS



“**NESS**” is a suffix that turns an adjective into a noun e.g. happy→happiness, not named after **a very famous Japanese game console**.

Wavetables are divided into two: **Tri** and **Pulse** waves.

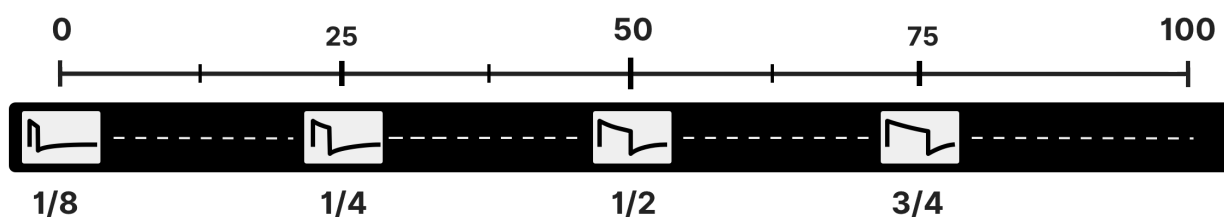
## NESS Tri



**NESS Tri** is a lo-fi triangle waveform. It has just 4bit(=16steps) resolution of voltage, producing characteristic digital noise in low notes.

All the 256 frames are used for subtle variations. Running through tables by LFO modulation can reproduce the unstableness of old hardwares. Differences between frames are so subtle that it may not be recognizable... But it does make a difference in some situations. Or even without wavetable modulation it has a vivid vibes of hardware sounds by virtue of the elaborately sculptured curves.

## NESS Pul



**NESS Pul**, after the fashion of the original hardware, has only 4 duty ratio namely 1/8(12.5%), 1/4(25%), 1/2(50%) and 3/4(75%). PWM is not available in this wavetable. 25% and 75% sound quite similar but not identical due to their unique curvatures.

Each duty of pulse wave has 64 frames of variations just like NESS Tri.

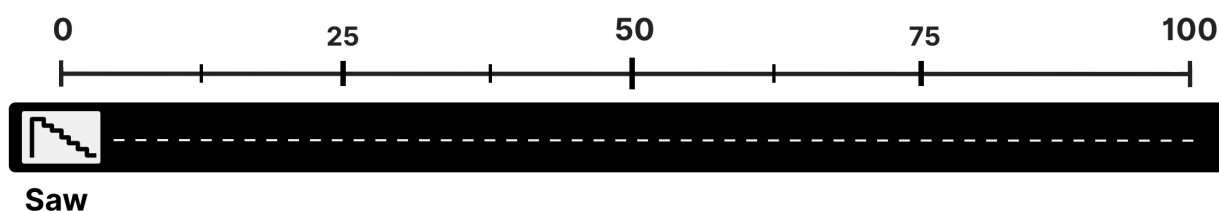
## VRC6



**VRC6** is literally modeled after an additional sound chip for Japanese game console. It has a pseudo-saw wave and a variety of pulse waves.

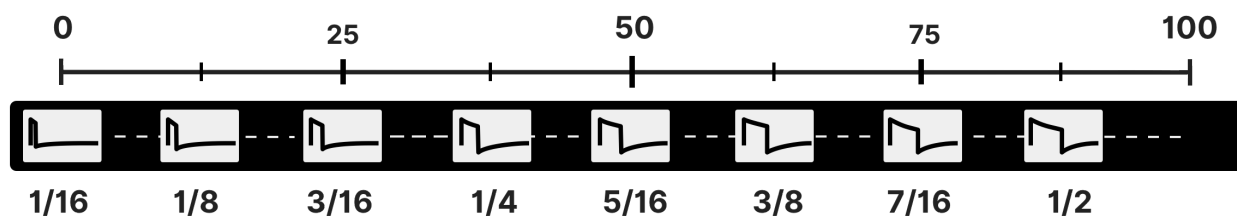
The games using this chip are very few and released only in Japan. The most famous one will be *Akumajou Densetsu (Castlevania III)*, where you can listen to boomy saw basses.

### VRC6 Saw



**VRC6 Saw** is a lo-fi sawtooth wave made up with just 7 steps. It sounds mean yet crisp--Great for Chiptune. Again, all the 256 frames are used for variations.

### VRC6 Pul



**VRC6 Pul** has 8 duty ratios from 1/16 to 8/16. Again, PWM is not available. Each duty has 32 frames of variations.

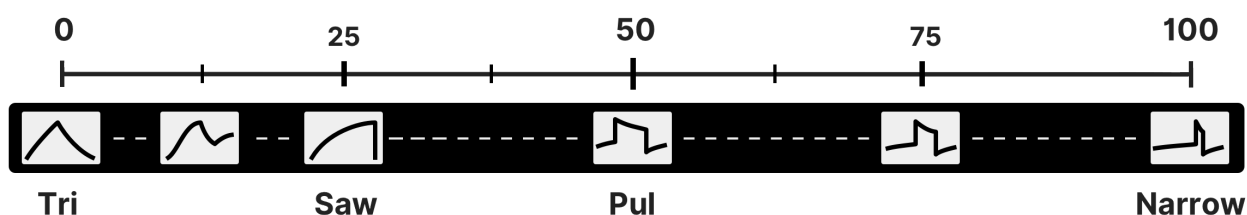
## SID



**SID** is literally modeled after the famous sound chip used in an **American 8bit home computer**.

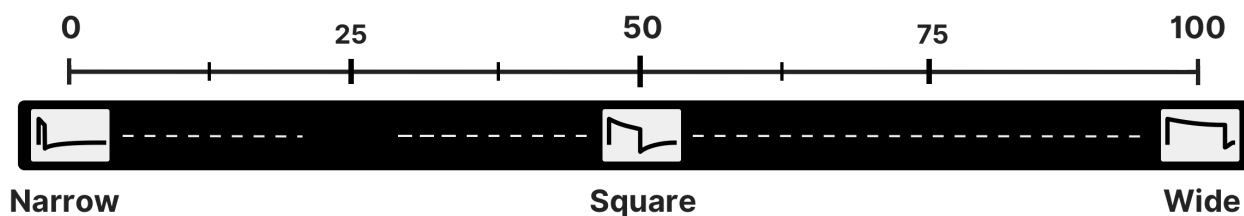
Tri and saw waveforms are much smoother here, almost like those of analog synthesizers.

## SID



**SID** contains all the basic waveforms—Tri, saw and pulse. Tri and saw here is much smoother than those of *NESS* or *VRC6*.

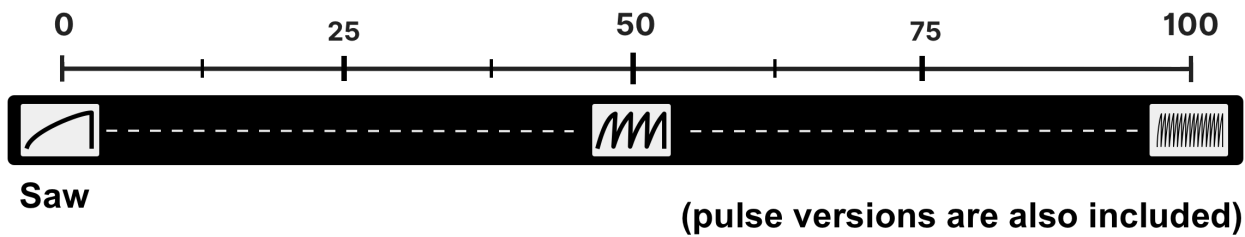
## SID PWM



**SID PWM** is a wavetable dedicated for pulse width modulation. The general *SID* wavetable also contains pulse waves but only 128 frames are allocated and the range of pulse width is limited. This one uses all the 256 frames for PWM.

## SID Sync

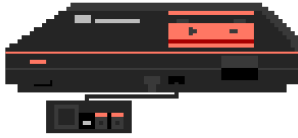
---



The original hardware has **hard sync** feature. A wavetable can imitate a specific situation of hard synced oscillator with pitch change.

**SID Sync~** waveforms reproduces hard-synced waveforms. WT position works as a tune of the master oscillator (Higher WT = higher synced pitch).  
There are 4 *SID Sync* wavetables with different base waveforms—Saw, square, 1/4 pulse and 1/8 pulse.

# OPLL Series



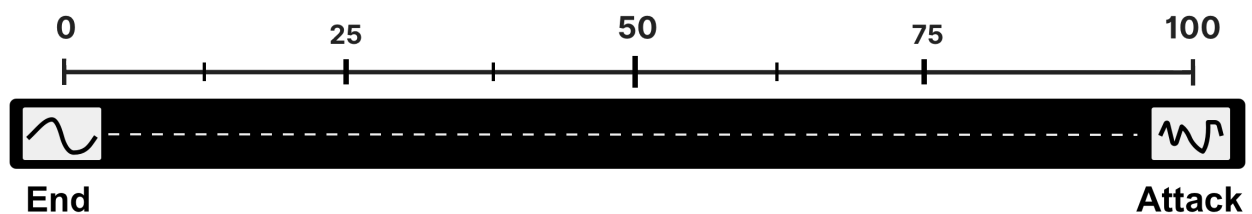
Wavetables in **OPLL** folder are the ones literally modeled after **OPLL(YM2413) FM** waveforms. As each wavetable mimics a specific instrument, they are not as versatile as generic waveforms like saw or pul.

OPLL originally has 15 preset sounds, all of which are recreated as wavetables in this product. But note that their sounds are *not* identical to the original ones. They sound very *close*, but not *equal* since the original parameters are in mystery.

Aside from the original 15 sounds, 11 additional wavetables using the same FM system are included.

## How 256 frames work

256 frames are used to replicate the **change of FM depth over time**. WT100 is the very start, the attack phase of an instrument and WT0 is the end of the sound transformation.



Most presets use an LFO or a FG to run through WTpos from 100 to 0, which is how HIVE imitates FM synthesis with just a wavetable.

## Tips: Understanding Original Hardwares

It is worth understanding the behaviors of original hardwares. Following such restriction can make a song sound more like a real tune of that age.

### The Number of Voices

Retro game sound chips has limited voices (the number of notes played simultaneously).

Chip	Voices
NESS	2 Pul + 1 Tri
VRC6	2 Pul + 1 Saw (+ 3 voices from "N")
SID	3 voices with arbitrary waveforms (Saw/Tri/Pul)
OPLL	6 voices + drums or 11 voices without drums

Polyphonic presets in 8bit Anthem basically follow these limitations e.g. Pads of NESS series set their maximum voices 3.

### Tunings

Retro game softs do not have the ability to directly point the exact frequency of a pitch. Pitches are generated from the calculation between the master clock and integer values programmed.

Since input values are integer-based, it causes tolerance—pitches are very slightly off-tuned from 12-tone equal temperament.

For example, pitch frequency in NES is calculated in the following way:

$$\text{Freq(Hz)} = \frac{\text{System Clock}}{16 \times \text{Input Value}}$$

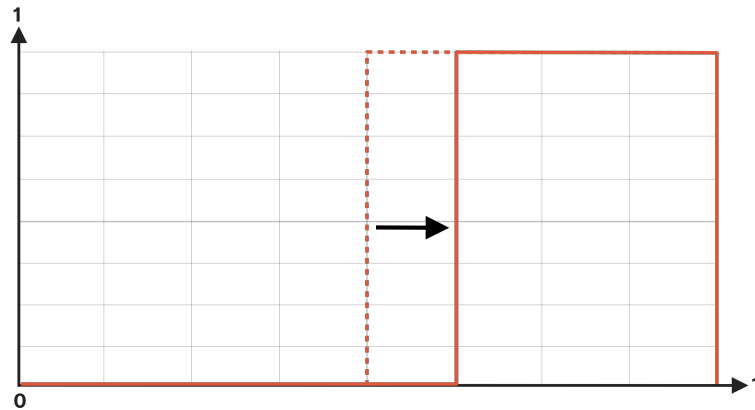
The diagram shows the formula for calculating frequency in Hz. The numerator is the System Clock, which is 1789772.7272. The denominator is 16 multiplied by the Input Value. The result is the frequency in Hz.

Presets in 8bit Anthem use **microtuning files** to replicate this formula. The very specific values to be substituted to **x** may vary according to softwares (regarding how to round decimals), so I choose simply to round off (round up 0.5 ↑, round down 0.4999 ↓).

# Miscellaneous Information

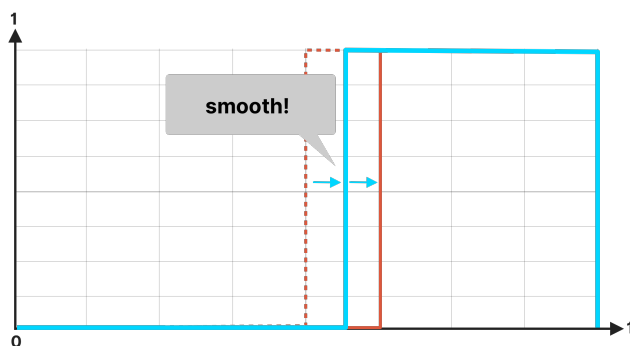
## About Interpolation

Remember that **Switch** interpol is recommended whenever there is a steep, solid vertical line moving horizontally e.g. PWM, Synced Pul.

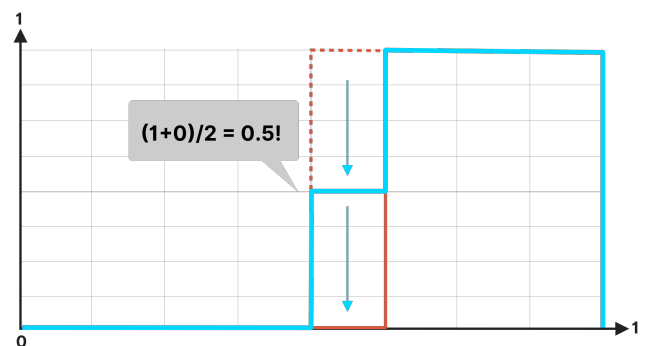


"Solid vertical line moves horizontally"

If you choose *crossfade* in these cases, unwanted interpolation happens. You would expect that HIVE intelligently connects between two, by smoothing its horizontal move. But the reality bites. Since crossfade calculates the *average voltage* of contiguous 2 frames, a strange step is produced.



What you expect to happen



What really happens

So *switch* interpol sounds more natural, paradoxically. Switching 256 frames is the best quality you can have for now.

## Why some of the WTs are converted to wav?

---

Firstly, note that the sound quality of uhm wavetables is **identical** to exported wav files. When you load a uhm file on Hive, it generates wavetables **on load**. On load it generates 256 frames of wavetable. So the difference is solely “storing it on computer as wav” or “generating it every time on load, for saving disk space”.

And since some of the wavetables are extraordinarily long, it takes some time to generate a wave, which is very stressful and sacrifices UX. So we choose to export as wav, while serving uhm files as an option for those who want to see the inside.

## What’s happening inside uhm? How do these codes work?

---

Read the [documentation](#) by U-he. Our codes are long, but primitive.

## Can I modify these uhm files?

---

Of-course yes. But please do not upload it public! —because they are part of product, part of what customers pay for. As to the details of the usage, please read the [Terms and Conditions](#).

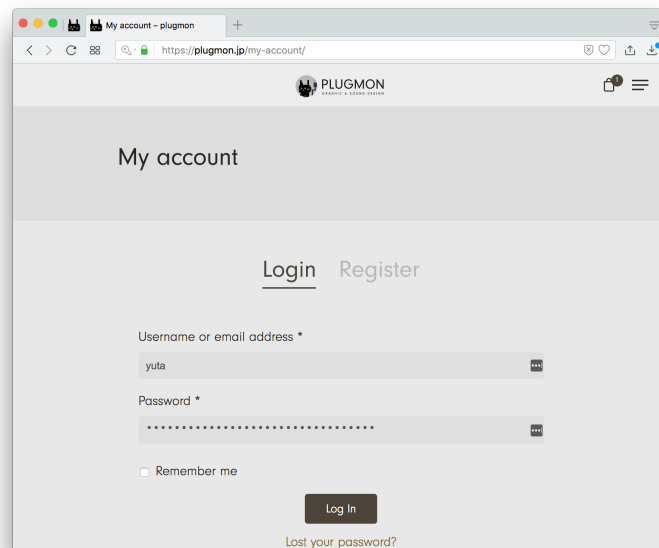


# Download & Update

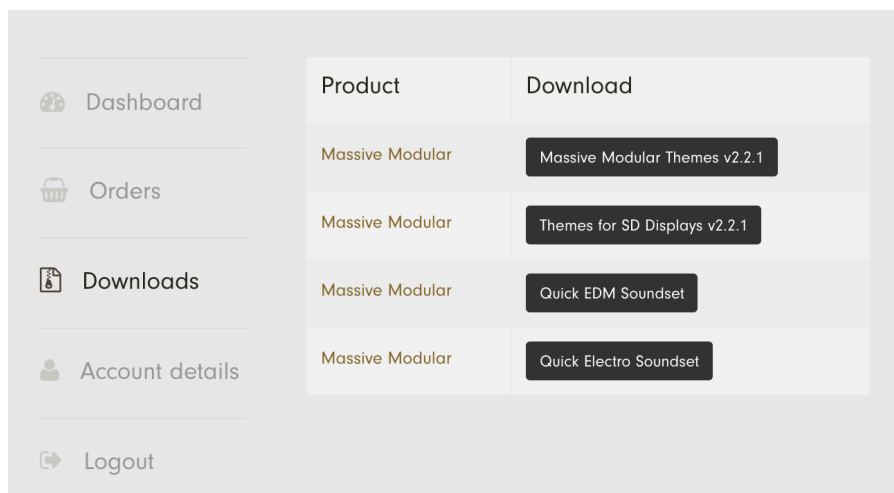
Re-download or update is provided via the website.

Your account page is : <https://plugmon.jp/my-account/>

Please login with your e-mail and password. **If you don't have account yet, you have to create one** (When creating, make sure that you enter the same address as you used on purchase).



After login, go to "**Downloads**" tab, where you can download your purchased products.



\*If you have any troubles, please [contact us](#).

## List of Wavetables

Below are the list of wavetables included in the package. All wav wavetables also has original uhm files from which wav files are generated. You can look into & edit these uhm files to customize the wavetables as you like!

NESS Pul.wav	OPLL 10 Synth.uhm
NESS Tri.wav	OPLL 11 Harpsichord.uhm
SID.wav	OPLL 12 Vibraphone.uhm
SID PWM.wav	OPLL 13 Synth Bass.uhm
SID Sync Pul 1-1.wav	OPLL 14 Acoustic Bass.uhm
SID Sync Pul 1-3.wav	OPLL 15 Electric Guitar.uhm
SID Sync Pul 1-7.wav	OPLL Clavi.uhm
SID Sync Saw.wav	OPLL Electric Bass 2.uhm
VRC6 Saw.wav	OPLL Electric Piano.uhm
VRC6 Pul.wav	OPLL Harp.uhm
OPLL Synth Sqr.uhm	OPLL Metallic Mallet.uhm
OPLL 01 Violin.uhm	OPLL Sax.uhm
OPLL 02 Guitar.uhm	OPLL Slap Bass.uhm
OPLL 03 Piano.uhm	OPLL Steal Pan.uhm
OPLL 04 Flute.uhm	OPLL Strings.uhm
OPLL 05 Clarinet.uhm	OPLL Synth Saw.uhm
OPLL 06 Oboe.uhm	
OPLL 07 Trumpet.uhm	
OPLL 08 Organ.uhm	
OPLL 09 Horn.uhm	