(https://www.noiseengineering.us/)

Basimilus Iteritas Alia Manual

Analog-inspired parameterized drum synthesizer

Overview

Basimilus Iteritas Alia is a parameterized digital drum voice with its roots in the analog world. At its heart, the BIA is a simple six-oscillator additive and FM synthesizer with waveform, harmonic spread, and envelope controls. An adjustable noise oscillator and an extreme take on a wavefolder round out its synthesis abilities. Its straightforward controls make sound design fun and performable: create sub-shaking kicks, snares, hats, and unique drum hits with just a few tweaks.

BIA is a classic percussion module but you'll be blown away by how much more it can do. Use it for leads, basslines, or build a whole drum kit with some external CV. Plus, the envelope output means you can use the BIA for shaping and modulating other parts of your patch.

Not only is BIA an incredible voice on its own, it's also part of the Alia oscillator platform. Use the included USB cable to connect to the `Customer Portal (https://portal.noiseengineering.us/?land=firmware)` and swap the firmware to any other Alia firmware, completely free, at any time.

> 💡 **Note**
>
> The Alia functions and sounds almost identical to original Basimilus Iteritas Alter. The Alia adds an envelope output, has a pitch encoder instead of a potentiometer, and has a pitch CV range of `-2 V to +5 V`; the orignal has a range of `0V to 8V`. The other interface features of the module are the same between Alia and original, and power/calibration instructions for the original BIA can be found in the **`Legacy`** section.

- **Type**: Universal drum synthesizer
- **Size**: 10HP Eurorack
- **Depth**: 1.5 Inches
- **Power**: 2x5 Eurorack
- **+12 V**: `105 mA`

- **-12 V**: `10 mA`
- **5 V**: `0mA`

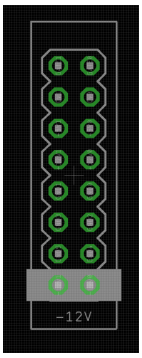
Basimilus - small foundation, from **Latin** *basos* "foundation" with suffix -lus "small"

Iteritas - repetitiousness - from Latin **itero** "repeat" with suffix **-tas** "state of being"

Alia - marine animal, or salty, from Greek **halia** "salty"

“More small salty foundation of repetitiousness”

Power



To power your Noise Engineering module, turn off your case. Plug one end of your ribbon cable into your power board so that the red stripe on the ribbon cable is aligned to the side that says `-12 V` and each pin on the power header is plugged into the connector on the ribbon. Make sure no pins are overhanging the connector! If they are, unplug it and realign.

Line up the red stripe on the ribbon cable so that it matches the white stripe and/or `-12 V` indication on the board and plug in the connector.

Screw your module into your case **before** powering on the module. You risk bumping the module's PCB against something metallic and damaging it if it's not properly secured when powered on.

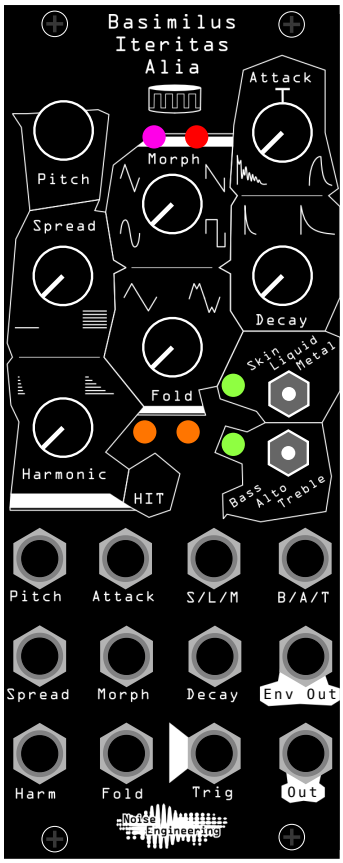
You should be good to go if you followed these instructions. Now go make some noise!

Noise Engineering modules are reverse protected. If you accidentally installed it with the red stripe up, simply remove the power and place it correctly.

A final note. Some modules have other headers -- they may have a different number of pins or may say "not power". In general, unless a manual tells you

otherwise, **do not connect those to power**.

Interface



Pitch

Encoder for adjusting pitch. Press and turn for coarse semitone adjustments, or just turn for fine tuning.

The `Pitch` input is calibrated for 1v/8va tracking.

Decay

The decay knob and CV input adjust the decay for all oscillators. The knob offsets the CV input.

Attack

The attack knob and CV input adjust the attack for all oscillators. When left of center, noise is added. When dead center, a classic analog-style pop is produced. When right of center, the knob slows the attack. The knob offsets the CV input.

Morph

The morph knob and CV control the waveform of all oscillators. This blends through sine, triangle, saw, and square continuously. The knob offsets the CV input.

Fold

The fold knob and CV control the infinifold section. For the first 3/4 of the range this sets the threshold of the folder. This folder will dynamically add multiple fold stages to maximize the amount of folding based on the fold threshold and signal amplitude. When the control is in the top quarter of its

range, a pulse train based on the signal is mixed in to give even more harmonic content. The knob offsets the CV input.

Harmonic

The harmonic knob and CV control the harmonic decay of the oscillators. When fully left, only one oscillator is audible. This simulates many simple analog bass drums. As the parameter is increased more harmonics blend in lasting longer periods of time. The knob offsets the CV input. Fully left the tone produced is a single harmonic tone. From there to the first quarter a second tone fades in. The remaining turning extends first the decays then the amplitudes of the other four harmonics.

Spread

The spread knob and CV control the frequency spacing of the oscillators. This allows the overtone series to vary from a purely harmonic to very inharmonic.

Skin/Liquid/Metal

Selects the synthesis algorithm. See tone generation for more information.

Skin (left): Six-oscillator additive for tonal sounds, snares, and synth stabs.

Liquid (middle): Six-oscillator additive with pitch envelope to add some extra kick.

Metal (right): A pair of 3-operator phase-modulated oscillators for producing metallic, noisy, and alien sounds.

Bass/Alto/Treble

Changes the pitch of the voice. Each switch position offsets the pitch by two octaves.

Trig

Triggers the envelope. The envelope has no Sustain phase, but any sort of gate or trigger will fire the envelope. Try it at audio rates, too! The Hit button manually activates the envelope.

Out

The output is a low-impedance audio source. The output varies significantly based on the parameterization as compensation for loudness occurs.

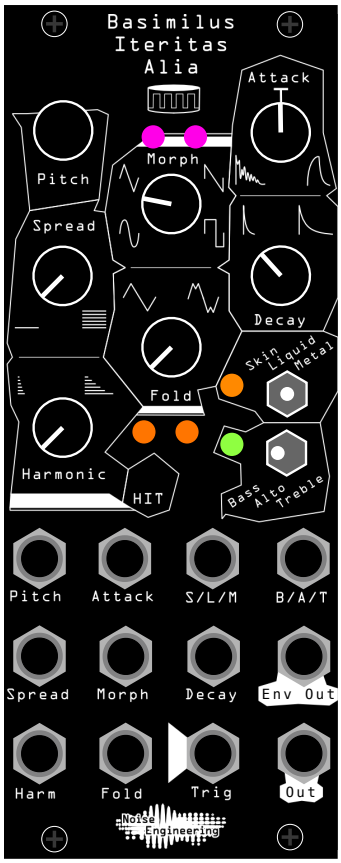
Env Out

An envelope output that mimics the envelope shape of BIA's internal envelope.

Patches

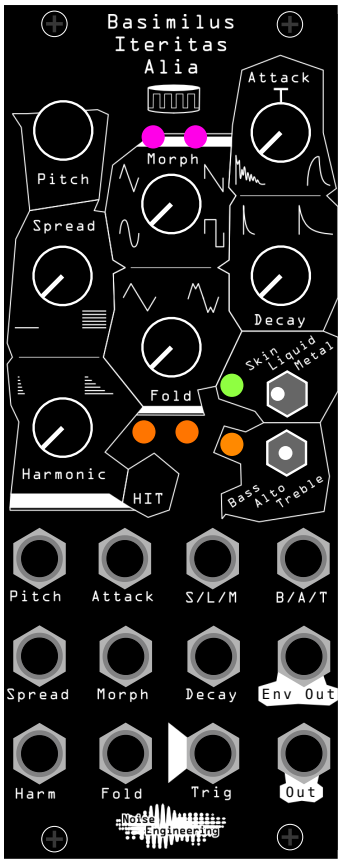
Simple Kick

Quick to patch and incredibly punchy kick. You'll be breaking windows in no time! Patch your trigger to Trig, and Out to your mixer. For extra fun, patch another trigger pattern to Decay to create accents! Attack and Decay shape a lot of the tone and length on this sound so play around with them to craft your kick to taste. Changing the position of Morph will also dramatically change the type of kick you get. Don't forget to tune it to the rest of your patch! Want a tom in your patch, too? Just turn up the pitch (or modulate it with CV)!



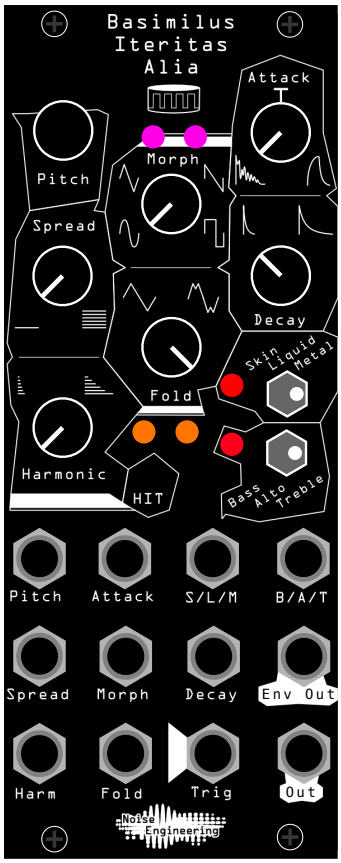
Simple Snare

Of course, you also need a snare with your kick. Patch your trigger to Trig, and Out to your mixer. Works great tuned high, low, or in between. Try the accent trick from the Simple Kick patch on this one too!



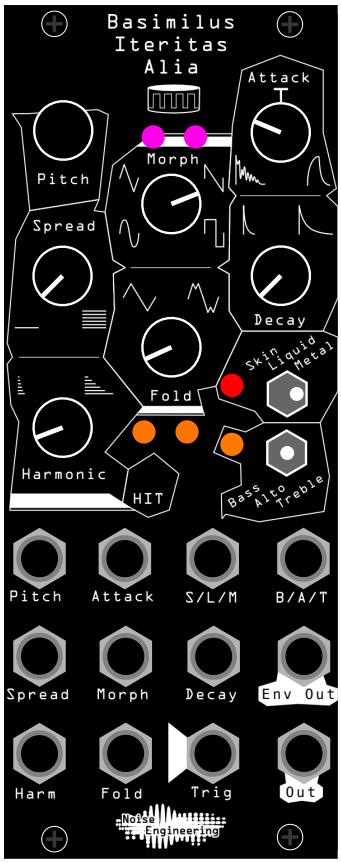
Simple Hat

Hats always add spice to a rhythm! Patch your trigger to Trig, and Out to your mixer. Just like the other drums, patch another trigger pattern to Decay to create accents! With some tweaking, this patch can also become a shaker or cymbal.



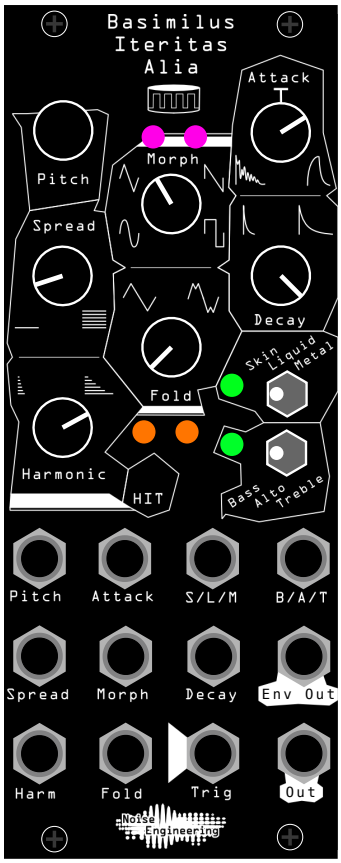
Simple Clap

Rounding out our drum machine with a clap! Patch your trigger to Trig, and Out to your mixer. Pitch changes the tone of the clap quite a bit on this one, so be sure to tweak to taste.



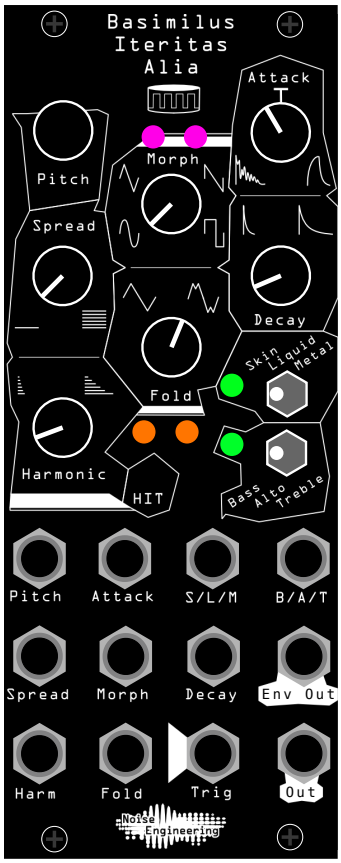
Supersaw

Synth basses and leads are also to be had with your BIA. Patch your trigger to Trig, and Out to your mixer. Patch a pitch CV sequence to the Pitch input and you'll be patching hits in no time.



Not quite FM bass

Think house basslines. Patch your trigger to Trig, and Out to your mixer. Patch a pitch CV sequence to the Pitch input. Fun!



Firmware swapping

Use our [firmware swap app (https://noiseengineering.us/portal/firmware/)](https://noiseengineering.us/portal/firmware/) to change your platform module's firmware at any time.

To get started:

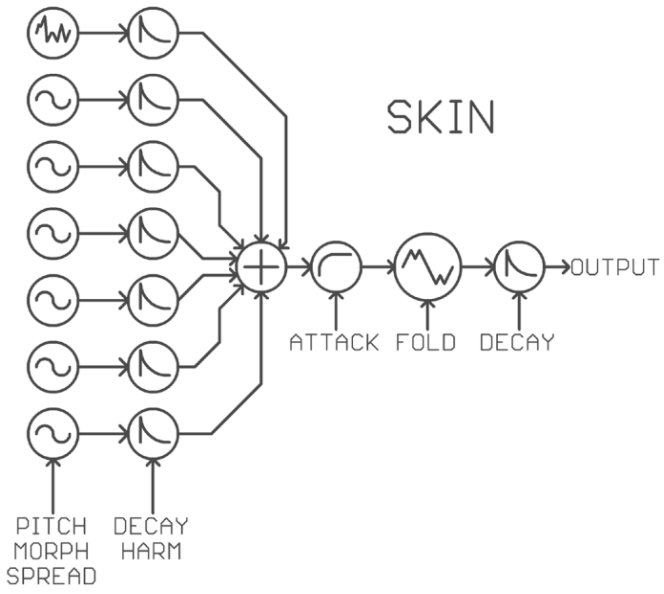
1. Turn off the power to your case and unscrew the module.
2. Remove the power connector on the back of the module.
3. Plug a micro USB connector into the port on the pack of the module, and the other end into your computer.
4. Follow the instructions in the [firmware swap app (https://noiseengineering.us/portal/firmware/)](https://noiseengineering.us/portal/firmware/).

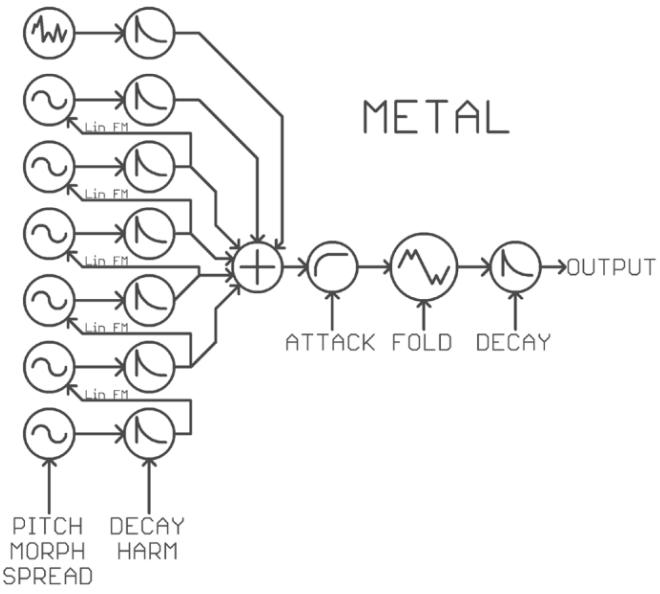
Tone generation

Basimilus Iteritas Alia uses six tonal and one noise oscillator in three configurations to generate sound. The `Skin` setting is a basic additive synthesizer meant to simulate instruments that have modes that do not interact. `Liquid` is the same as skin but with a pitch envelope for all oscillators. The first oscillator frequency is determined by the pitch input. The `Metal` setting modulates the oscillators by each other to simulate instruments that have a lot of modal interaction. The `Spread` control adjust the pitch (relative to the base pitch) of the other five oscillators. Each oscillator has an individual envelope that is controlled by the `Attack`, `Decay`, and `Harm` controls. The noise envelope is also affected by the `Attack` knob.

The oscillators are summed and then the `Attack` envelope is applied to the sum. This then feeds into a threshold-reflection folder with amplitude compensation and the ability to dynamically add more fold stages. At very high settings the fold will add in an exponentially decaying pulse at the local minima and maxima of the signal to add a gnarly buzz.

The final step is another envelope. This envelope is derived from the overall shape of the six oscillator envelopes. It adds back in the dynamics lost by folding so the output is remains punchy under the most extreme folding.





Input and output voltages

Alia's trigger input has a threshold around `+1.8 V`.

Its modulation CV inputs have a range of `0 V to +5 V`.

Its pitch CV input has a range of `-2 V to +5 V`.

The envelope output has a range of `0 V to +5 V`.

The audio output varies depending on settings, and can reach a maximum of about `14 V peak to peak`.

Calibration

Alia features an autocalibration system. The modules are autocalibrated and tested at the factory, but should you feel you need to recalibrate, just power the unit on with nothing patched to the Pitch CV input. The module will calibrate itself during startup.

Variable sample rate

Basimilus Iteritas Alia uses a sample rate that is a multiple of the fundamental (lowest) oscillator frequency. This moves alias power that is a multiple of the fundamental to be mapped to a multiple of this tone, therefore making the aliasing align with the harmonics of the tone. This works well for settings with a strong harmonic structure (spread fully clockwise or fully counter-clockwise) and adds unique aliasing character for other tones.

Genesis & design notes

The following was written by Stephen when the original BIA was released.

Basimilus Iteritas started out as something very different than where it ended. I had been reading Stefan Bilbao's *Numerical Sound Synthesis* (https://dl.acm.org/doi/book/10.5555/1823189) and was thinking about building a drum simulator out of a set of numerical oscillators. I looked at spectrograms of many kick drums and settled on a one oscillator per drum mode possibly adding interesting cross-mode interactions.

A few hours into a software prototype convinced me this was not going to be a simple program. Numerical oscillator implementation has a number of practical problems and pitfalls that I was falling deeply into. Taking this difficulty as an opportunity to reflect, I asked myself, "What would analog do?" I answered "construct a similar sound out of multiple easy to implement elements." This led me directly to a simple wavetable additive architecture with exponential envelopes and noise. Each oscillator is an analog of one mode of a drum. There is a touch of irony as the analog oscillators used in actual analog kicks are actually more closely related to the numerical methods than the additive wavetable architecture that was finally settled on. This new architecture was very suited to parameterization from musical terms rather than the terms that numerical models demand. Even such basic things as pitch can be somewhat tricky with the methods in Bilbao.

The hardware development for Basimilus Iteritas went very smoothly. Only three PCB revisions exist and the last two were almost identical apart from some minor changes to the ADC overvoltage protection.

The CPU daughter-board was developed to be used in a wide range of future modules. It uses the XMOS XS1 (https://www.xmos.ai/xs1/) processor which is a great blend of flexible, powerful and affordable. Basimilus Iteritas was written in XC and C using the XMOS IDE. The algorithm was prototyped in C# and C as a windows application to quickly iterate on the parametrization.

The internal signal chain is 8.24 fixed point which is quantized to 16-bit for the DAC (TI 8411 (https://www.ti.com/product/DAC8411)). Basimilus Iteritas uses a dynamically adjustable sample rate to help move alias power to be at frequencies related to the fundamental pitch of the drum and therefore be more musical. The oscillators are essentially wavetable though they are evaluated on

the fly as this is needed for the `Morph` knob. They have a period of 65536 samples but are decimated by a different amount depending on the octave of the pitch. The noise generator is a simple linear congruential generator decimated by octave similar to the waveform oscillators. The `Metal` setting is classic 80's "who cares about aliasing?" frequency modulation in all of its noisy glory.

Analog drums sound awesome in folders so a folder was an obvious add. Once advantage of digital folding over analog is that additional sections are essentially free. Basimilus Iteritas adds as many fold sections as will still continue to fold to maximize the amount of produced harmonics. It as well cleanly compensates for the volume changes that occur during folding. The final touch was to re-apply the overall envelope to the signal after the folder which gave back a lot of the dynamics that are lost when folding to produce a kick that still has some percussive umph after being massively folded.

The real complexity of sound comes through the `Harm` and `Spread` knobs. `Spread` is quite simple in implementation as it adjusts the intervals between the drum modes from the harmonic series to the prime series. `Harm` is quite a bit more complicated as it adjusts the decays and amplitudes of the oscillators to produce a wide variety of tonal structures. The goal was to be able to produce both one and two tone analogish drums, drums with a lot of modal power that decays quickly and drums that have long decaying modes.

After two years of Basimilus Iteritas being in the wild, a number of design improvements had become obvious. As part of ongoing engineering improvements we needed to change the CPU board that was used on the original Basimilus and this became an opportunity to do a more in-depth revamp.

The single biggest change was the addition of Liquid mode. Pitch envelopes are a pretty obvious thing to do with something like the Basimilus and a lot of people do it externally. Chris Randall sent me a Neuron to play with which made me very quickly realize how much fun a pitch envelope would be. Within a few hours this was prototyped and was slated for inclusion on the upcoming hardware revision.

I hope you have as much fun with Basimilus Iteritas Alter as I had designing it!

Enter Alia

All good things must come to an end. Thankfully, that time hasn't come for BIA yet, and Basimilus is still kicking.

When the manufacturer of the processor used in the original BIA announced that it would no longer be available, we rushed to find an alternative. There were many, many challenges but we were able to make some improvements, too: the new encoder means more precise tuning, and the envelope output adds a new level of patchability. Bringing the Iteritas to a platform was also exciting for us: we love being able to bring new ideas to users quickly and easily.

Warranty

We will repair or replace (at our discretion) any product that we manufactured as long as we are in business and are able to get the parts to do so. We aim to support modules that have been discontinued for as long as possible. This warranty does not apply to normal wear and tear, including art/panel wear, or any products that have been modified, abused, or misused. Our warranty is limited to manufacturing defects.

Warranty repairs/replacements are free. Repairs due to user modification or other damage are charged at an affordable rate. Customers are responsible for the cost of shipping to Noise Engineering for repair.

All returns must be coordinated through Noise Engineering; returns without a Return Authorization will be refused and returned to sender.

Please contact us (https://noiseengineering.us/pages/contact) if you think one of your modules needs a repair.

Special thanks

- Kris Kaiser

- Shawn Jimmerson
- Christopher Randall
- William Mathewson
- Mickey Bakas
- Tyler Thompson
- Alex Anderson

Legacy

The information in this section only applies to the original Basimilus Iteritas/Basimilus Iteritas Alter hardware, not the Alia.

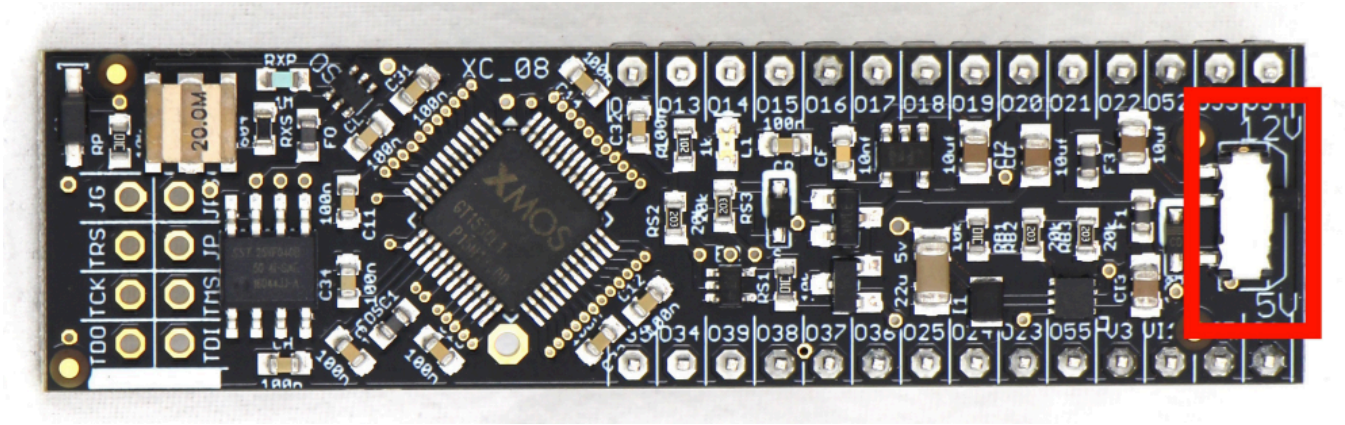
Legacy tuning calibration

Basimilus Iteritas Alter comes pre-calibrated and should not need adjustment. If the trimpot gets bumped and needs adjustment, follow this procedure to calibrate your module. Pitch calibration is controlled by an linear resistor-divider network. To calibrate the tuning, attach a voltmeter (preferably 4 or more digit) to the test points `TPCV` and `TPGND` on the rear panel and adjust the trim pot.

The voltage measured should be `5/16` (`.3125`) times the input voltage applied to the CV input. A reasonable way to tune the scale is to use an adjustable voltage source to generate `4 V` then adjust the tuning trim until the test points read `1.2500 V`. Basimilus Iteritas Alter can also be tuned using a reference supply capable of generating a `1 V` difference and using a stroboscope such as the Peterson 490 to tune to an octave interval. This is method is preferred to the meter-only method.

Legacy voltage supply

Basimilus Iteritas Alter can run its processor on the `5 volt` eurorack power rail to reduce noise and load on the `12 volt` bus. Gently push the switch tab in the direction of the desired rail to use.



Last updated on October 22, 2025