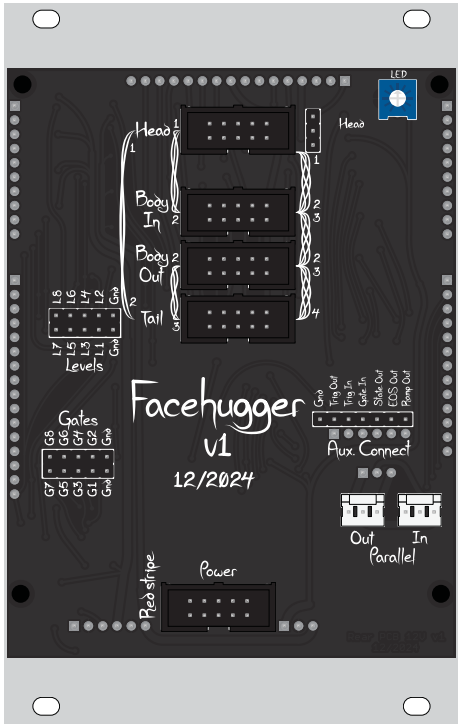
flight of harmony

Facehugger

Eurorack Control Module
v1



Specifications @ ±12VDC

Width	16hp	
Depth	Module only:44mm / 1.7" With cables 53mm / 2.1"	
Supply Voltage	±12VDC	
Supply Current (max draw @ ±12V)	I_{+12V}= 150mA	I_{-12V}= -90mA
Max. Input Voltage	±5V / $10V_{P-P}$	
Max. Output Voltage	±10V / $20V_{P-P}$	
Rear Gate out	10VDC	
Max. Rear Level out	±10V / $20V_{P-P}$	
Input & Ouput (I/O) coupling	Direct	
Output Impedance	1kΩ	
Control Voltage (CV) inputs	±5V / $10V_{P-P}$	
CV input coupling	Direct	
CV input impedance	100kΩ	

FYI

As with all f(h) products, the Facehugger was engineered towards maximizing functionality while keeping cost as low as possible. If some aspects of the unit seem awkward, it is most likely due to this. The goal is to make unique, useful, enjoyable, and affordable instruments, not just hoover[1] out your bank account.

1) Hoover is a company that manufactures vacuum cleaners.

Also, it's analog, which means temperature drift and other quirks are features, not issues, so enjoy the erraticity! Also, because of this, all output values in this manual are approximations and may vary from unit to unit, and even from slider to slider[2].

And remember: every instrument has its quirks and unexpected aspects, so RTFM[3]! All the way through! Specific quirks are mentioned in the description of the particular feature they apply to, so please read this through before emailing.

What is it?

It started out as an analog quasi[4]-arpeggiator – you send it a trigger and it outputs a sequence of control voltages that can be shifted up or down *en masse* to follow a tune – but it needed a bit more to be worth making. Now, it's an arpeggiator, burst generator, oscillator, arbitrary waveform generator, envelope generator,and just all-around control module, i.e., a thing that tells other things what/how/when to do things.

What does it do?

Facehugger outputs a signal based on the position you set the slide potentiometers to, one at a time, at a speed and direction – left-to-right or right-to-left – set by the other controls, to then control other modules, such as to the pitch CV input of a VCO[5]. It can run single-shot, from a trigger, a longer duration from a gate signal, or indefinitely from the "Run" setting.

There are also several headers on the rear of the module for upcoming expansion modules, and whatever else you may want to use them for.

That's the very basic overview. A major focus when developing the Facehugger was to be able to extend this in as many directions as possible and enable whatever ridiculously convoluted and intricate arrangement might be desired. To do this:

Facehugger modules can be chained and paralleled indefinitely[6].

Chaining means connecting in series: as soon as one stops, it tells the next one to start, and so on until it loops back to the beginning (the head unit). The output of each chained unit is available from the head unit, as well as each unit outputting its own sequence. More units gives you more stages – 8, 16, 24, etc. – and each group of 8 can have its own variances. More details in the section "Chaining and Paralleling" on page 9.

Paralleling means operating simultaneously: each unit in parallel is synchronized with the head unit and runs at the same speed. Here, the only synchronization is the speed at which they run, everything else can be varied. Paralleling is great for making chords and other synchronized effects. More details in the section "Chaining and Paralleling" on page 9.

2) The option being paying $100 or more *per slide pot.*
3) RTFM = Read The F*cking Manual!
4) "quasi" because rules: an arpeggio is specifically the notes of a chord played separately, but there is no feature to maintain relationships between notes here because no.
5) VCO = Voltage-Controlled Oscillator
6) At the time of writing this, I have had four units chained with three units paralleled simultaneously.

Level slide potentiometers
Adjust output level of each stage

Operation modes
↘ direction left to right, output positive only

→ direction left to right, output bipolar

← direction right to left, output bipolar

Internal timing ramp Curve shape
◜ logarithmic/decelerating last stage longest

／ linear, all stage lengths equal

◝ exponential/accelerating, first stage longest

External timing waveform input
replaces internal ramp
signal value determines active stage
expects -5V to +5V signal. Effect varies with Operation Mode

Speed
Duration of sequence.
CV input accepts bipolar signals

Root
Sets 0V position of sliders/ shifts octave of output
CV input accepts bipolar signals

Scale
Scales magnitude of output
CV input accepts bipolar signals

Signal output
Outputs values of sliders based on all settings.
from 0 –> +10VDC to ±10VDC

Glide
◜ logarithmic/curved

— no glide (center off)

／ linear (best for fast envelopes)

Stages
Number of stages in one cycle.
1-8 possible stages

Run modes
Run free-running.

T Trigger only
Runs once and stops
ignores trigger length
derives trigger from any sharp-edged input

T\G Trigger with Gating
Runs as long as button is held or input is high (+5V or higher)

fig. 1: Facehugger front panel controls and jacks.

Front panel controls and jacks (fig. 1)

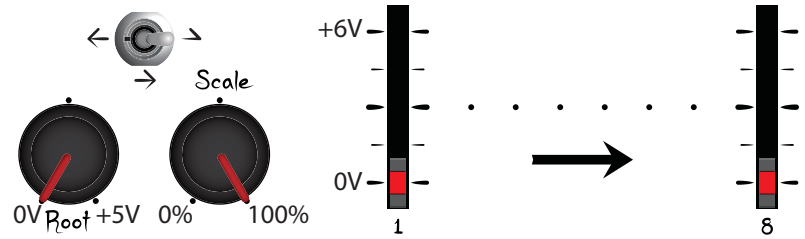
Selection switches:

Run Mode: Selects between

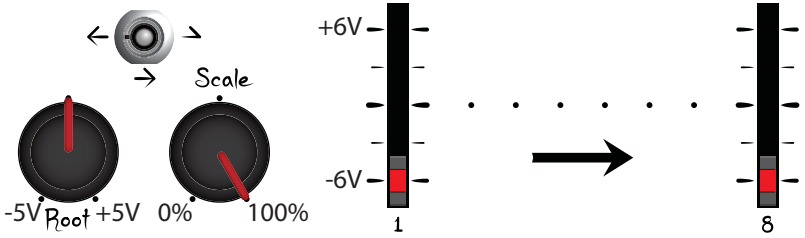
- Run: Free-running
- T: Triggered. Runs once when triggered via either the pushbutton or from any sharp-edged transition from the Trigger jack.
- T\G: Gated. Runs as long as button is held or trigger input is > +5V.

Operation Mode: Selects output range and cycle direction[7]:

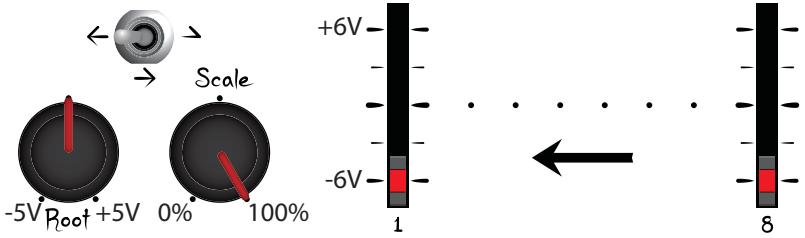
- ↘: direction left to right (1->8), output positive only.



- →: direction left to right (1->8), output bipolar.



- ↙: direction right to left (8->1), output bipolar.



Curve shape: Direction and amount of curvature applied to internal timing ramp.

- ◜: Logarithmic/decelerating, first stage shortest, last stage longest.
- ╱: Linear, all stage lengths equal.
- ◝: Exponential/accelerating, first stage longest, last stage shortest.

Due to the technique used to implement the curve feature, each **Curve** position runs at a moderately different speed, with Linear as the slowest, then Logarithmic, and Exponential being the fastest. Log and Exp are very close to each other, at about 135% of linear.

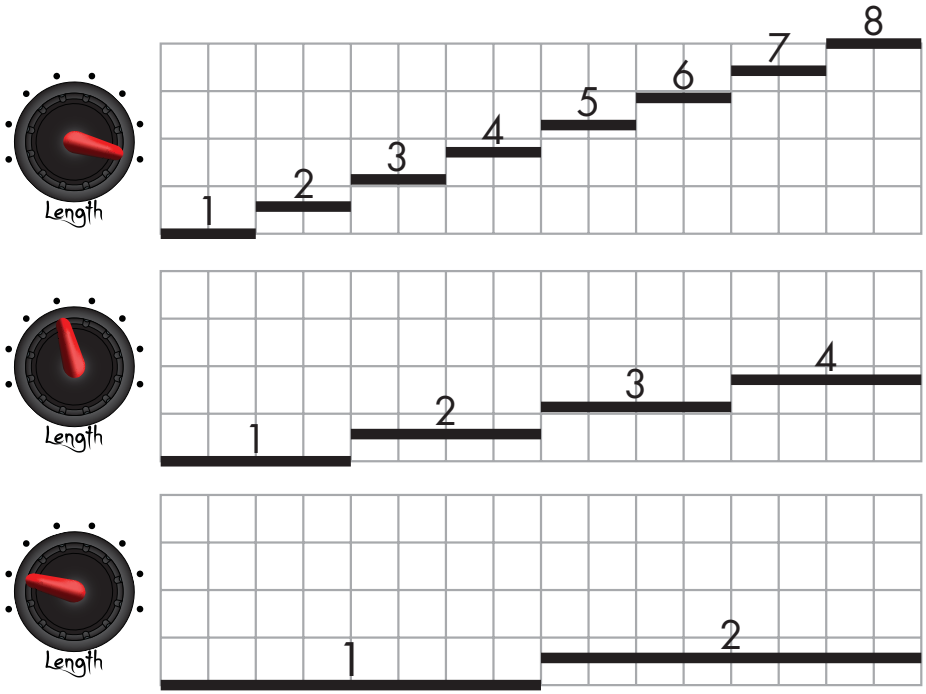
Glide: Type and amount of glide /portamento applied to output.

- ◜: Logarithmic/curved.
- · : No glide (center off)
- ╱: Linear (best for fast envelopes)

Knobs:

Stages: Number of stages per sequence/cycle, from 1 to 8. Sequence length is indicated by the numbered LEDs.

The stage **Length** sets how many stages happen within one sequence but does not affect the total sequence duration, so the length of each stage varies with how many stages there are:



Curve: Amount of curvature applied to internal ramp (see p. 14).

Speed: Duration of sequence/how fast it runs.

Root: Sets baseline of output (see p. 15).

7) Values shown are approximations.

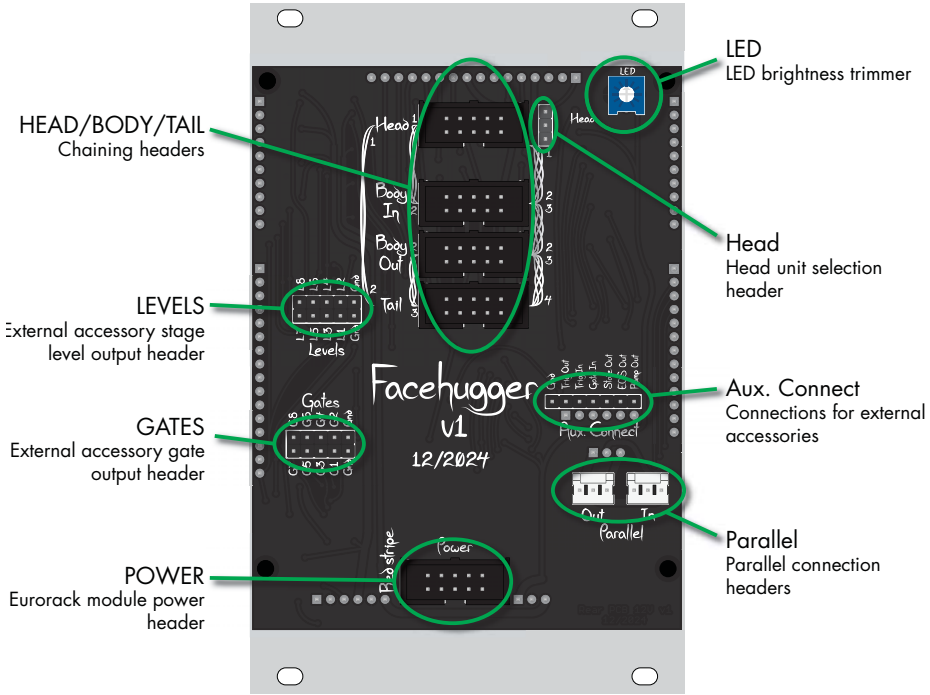


fig. 2: Facehugger rear view

Scale: Adjust amplitude of output, from 0 to +10V or ±10V, depending on Operation Mode setting (see p. 15).

Glide: Amount of Glide applied to output (see p. 14).

Glide will always have *some* effect on the output, even when turned fully counter-clockwise. The amount increases with speed, from negligible at slow speeds to very dominant at audio rate and above.

Jacks:

Note: all jacks accept any signal within eurorack standard levels.

Ext In: External signal input. Inserting a plug here disconnects the internal timing ramp and replaces it with whatever is present on the input.

Speed: CV control of internal ramp speed.

Root: CV control of the Root level.

Trigger: External trigger and Gate input.

Scale: CV control of output amplitude.

Out: Signal output.

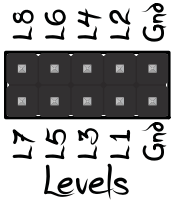
Rear connections & controls (fig. 2)

HEAD/BODY/TAIL: Chaining headers (see p. 10).

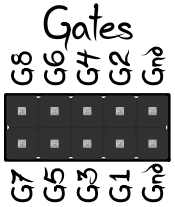
Head: Head unit selection header (see p. 11).

Aux. ConnectAux. Connect: Connections for external accessories (see p. 12).

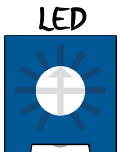
Parallel: Parallel connection headers (see p. 12).



Levels: Stage level output header for external accessories -10VDC to +10VDC. Outputs the value of each stage at all times, running or not. These outputs are unbuffered; a direct connection to the signal before it enters the root and glide section (see p. 18). Unbuffered use will affect the front output.



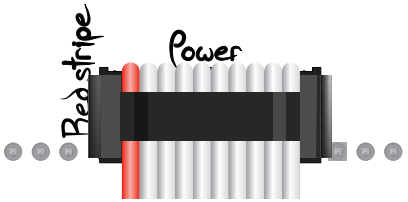
Gates: Gate output header for external accessories[8]. Outputs a +10VDC gate signal for each active stage. Maximum output sink/source is 10mA, so use as a logic signal source only. It is not designed for driving a circuit.



LED: Trim potentiometer to adjust the global LED brightness. Clockwise for brighter, CCW for dimmer.

Power: The power connector header is a 2x5/10-pin shrouded box header which accepts the standard Doepfer power cable. This header style is polarized, meaning the connector can only be inserted one way, to prevent connecting the power backwards and damaging the unit (this assumes that you are using the cable supplied with the module).

Looking at the rear of the module, the negative supply (red stripe) is on the left, positive supply is on the right.



8) Why, oh why, are the pins in such an OCD-mocking arrangement on these two headers? Because that was the arangement that kept the traces as short as possible since the PCB is freaking packed and didn't have room for me to be a proper autist.

USING THE THING

- Rack it, power it.
- Connect **Out** to the CV input of something.
- Randomly arrange the sliders.
- Turn **Scale** to midway.
- Turn **Speed** to about 10 o'clock
- Press the trigger button, see what happens.
- Now mess with all the settings and see what else happens.

Basic operation

- Module is triggered, gated, or set to **Run**
- Output is engaged and sequence begins
- Sequence ends and device either:
  - Restarts at beginning of sequence if set to **Run** or is gated.
  - Stops and disengages output

Output engage/disengage

The output of the Facehugger disengages/disconnects from the jack when not running so it is invisible to the rest of the system when not running. This prevents the setting of the initial state (the first slide pot) from being at the output and affecting connected devices when not desired.

Chaining and Paralleling

NOTE: Please only use the supplied chain and parallel cables. Facehugger has only been tested in the following configurations, and attempting other arrangements using custom cabling will likely cause spontaneous vomiting and implosion, as well as destroying the devices. Cats will judge you, it will be bad.

Chaining: Connecting multiple Facehugger modules in series so all run sequentially. The first module is the **Head** unit, the last module is the **Tail** unit, any in between are **Body** units. Behavior of each unit:

Head

- Behaves normally when running
- Output of each chained unit appears at **Out** as they run.

Body/Tail

- Behaves normally when running except:
  - **Trigger** and **Gate** button press or input signals are rerouted to **Head** and restart full sequence regardless of place in chain.
- Output of unit still appears at **Out** as it runs and disengages when not running[9].

Paralleling: Connecting multiple units in parallel so all run together at the same speed and each stage is synchronous. The first module is the **Head** unit, all subsequent modules are **Tail** units. Behavior of each unit:

Head

- Behaves normally

Tail

- **Speed** and **Curve** controls and inputs are ignored.
- **Operation Mode**s still work normally
- Internal ramp still affected by Speed & Curve and is available at **Ramp Out** pin on **Aux. Connect** (see p. 12).

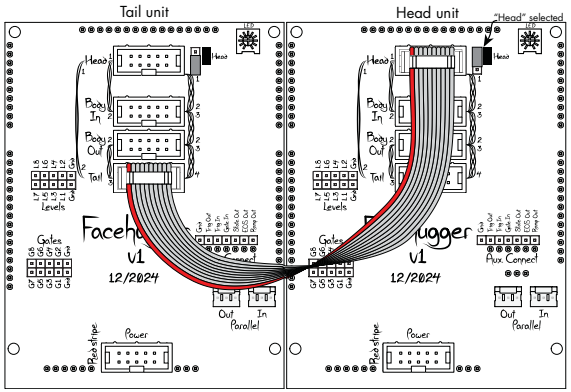
You can chain and parallel as many Facehuggers as you want in any combination, but there can be only ONE connection between any two devices. DO NOT CONNECT TWO DEVICES AS CHAIN AND PARALLEL AT THE SAME TIME. This has been tried, it does not go well at all. If you let the smoke out, I will laugh at you and think very unkind thoughts while fixing them.

HEAD/BODY/TAIL

These headers determine what each module does and its place in the chain. On most f(h) products, I say plug anything into anything, it'll be fine, but not here. THESE ARE NOT FOR EXPERIMENTATION OR ADAPTING, a wrong connection can destroy the modules. The acceptable connections are:

Two Facehuggers

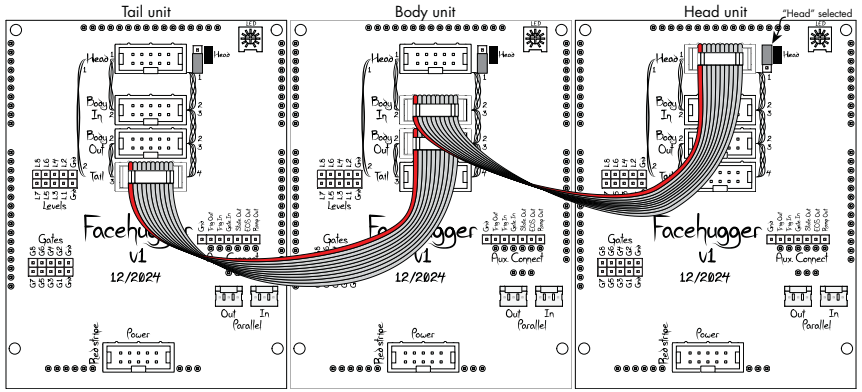
- **Head** to **Tail** (required)



9) While this does fall under "behaves normally", it felt like it was worth stating explicitly.

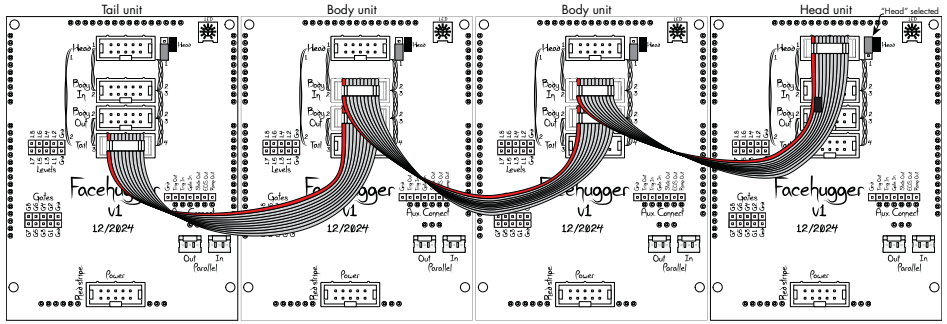
Three Facehuggers

- **Head** to **Body In** (required)
- **Body Out** to **Tail** (required)



Four or more Facehuggers

- **Head** to **Body In** (required)
- **Body Out** to **Body In** (required)
  - Repeat for any additional units (required)
- **Body Out** to **Tail** (required)



Head

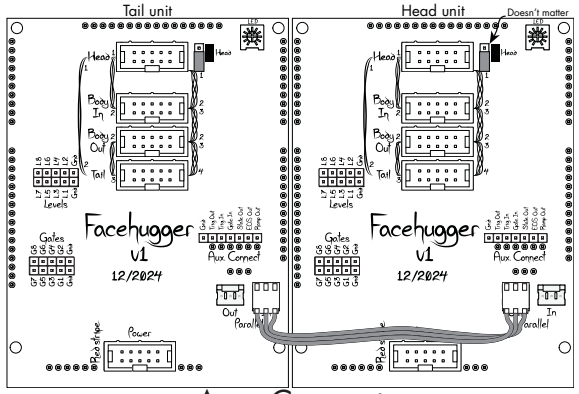
A bit of a kludge, but this jumper is needed to keep the output of the head unit open while it is waiting for the other units to finish and mirror their outputs. Place the jumper on the top two pins of the header for the **Head**, and on the bottom two pins for all **Body** and the **Tail**.

Parallel

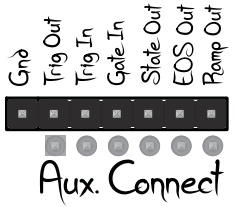
These headers determine what each module does and its role in paralleling. Again, THESE ARE NOT FOR EXPERIMENTATION OR ADAPTING, a wrong connection can destroy the modules. The acceptable connections are:

- **Parallel Out** to **Parallel In**

That's it. Don't try anything else. But you can keep doing it with as many as you want.



Aux. Connect



Okay, *these* you can play with, that's what they're for, as well as some expansion modules that are in planning.

Gnd: Ground pin, in case it is needed.

Trig Out: +10VDC output signal. Relays any trigger from the Trigger pushbutton or the Trigger input jack.

Trig In: Accepts any sharp-edged signal ≥+5V.

Gate In: Accepts any signal ≥+5V.

State Out: +10VDC output signal. Goes high while internal ramp is running, 0V when idle.

EOS Out: +10VDC, 1mS output signal. EOS = End Of Sequence. Outputs a trigger pulse at the end of the last stage of any sequence.

Ramp Out: 0 -> +5VDC ramp output. Outputs a copy of the internal ramp. Not affected by the input jack.

Fun note: The internal ramp is always available at the **Ramp Out** pin as long as the module is triggered, gated, or in **Run** mode, even with an external signal in the **Ext In** jack, being Paralleled, or Chained. The ramp is also always affected by the settings of **Curve** and **Speed**.

Patches and ideas:

Contact me at the email below and come check out the f(h) Discord server for examples, demonstration videos, and discussion on what you can do with the Facehugger!

Stuff

Eternal gratitude to the many who supported the development of the Facehugger, it would not exist[10] without your backing and patience.

A big thank you to those who have sent in suggestions and comments, keep them coming!

Comments, samples, suggestions, complaints to: flight@flightofharmony.com

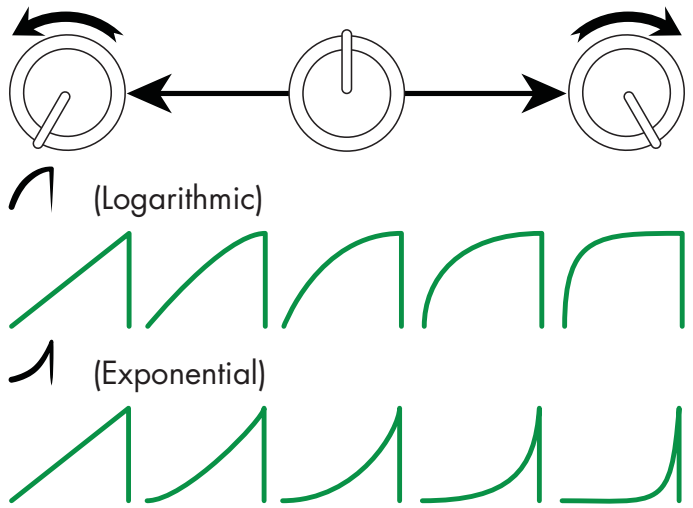
We have a Discord server! Contact me at the address above for an invite link.

http://www.flightofharmony.com

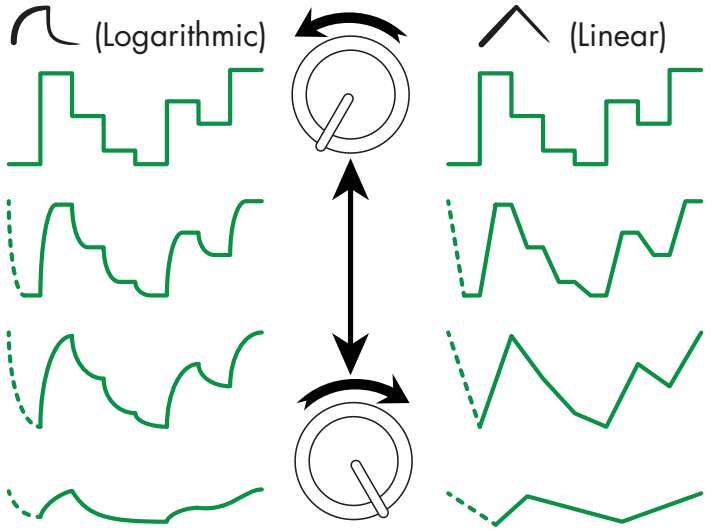
f(h)

10) Seriously. Facehugger development has taken over three years. I would have shelved this thing long before if I didn't have y'all quietly waiting, watching...

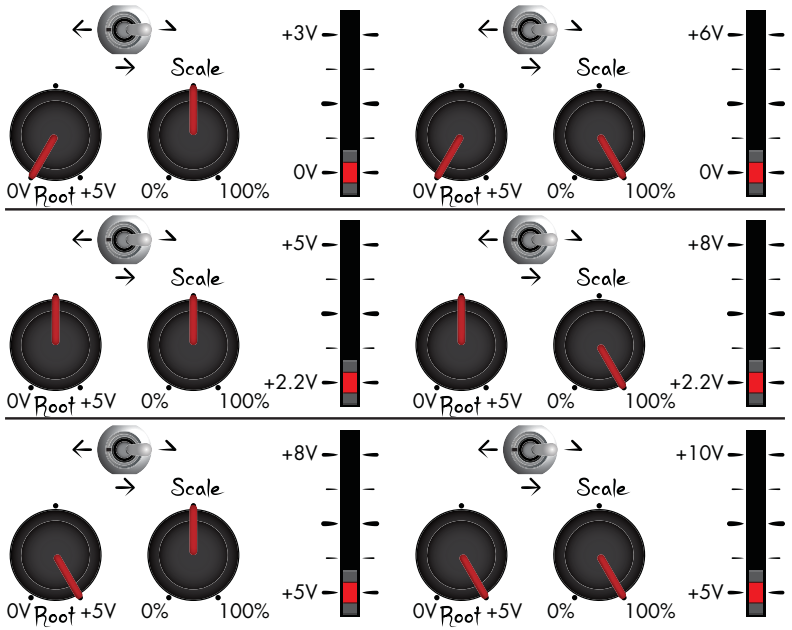
Effect of Curve potentiometer on internal ramp wave:



Effect of Glide potentiometer on output:

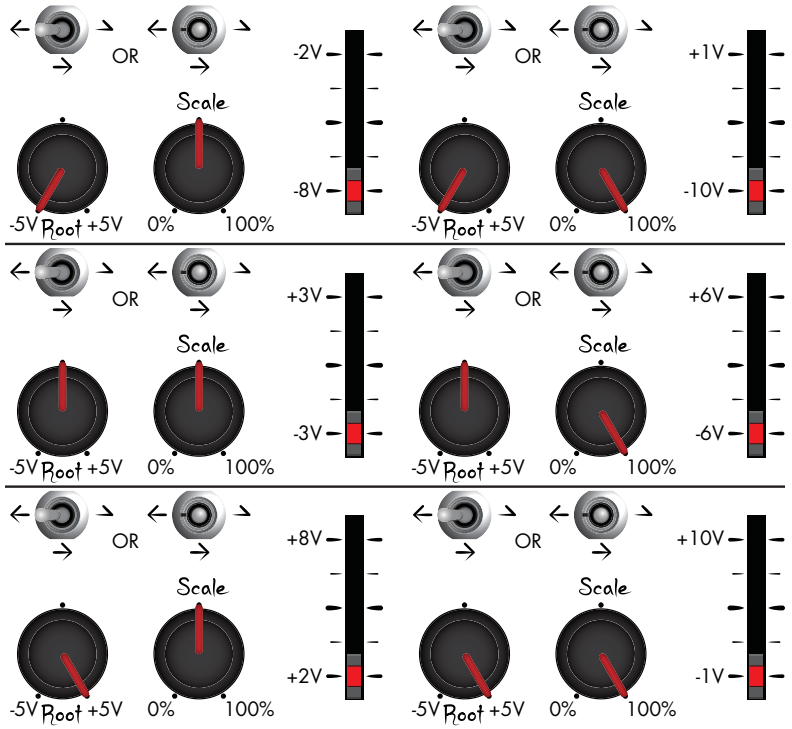


Effect of Root and Scale on stage level in → mode



All values are approximate and may vary.

Effect of Root and Scale on stage level in ← or → mode



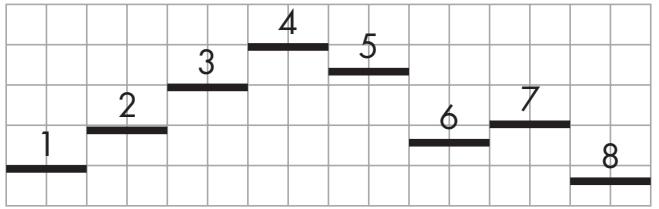
All values are approximate and may vary.

Stage length adjusting

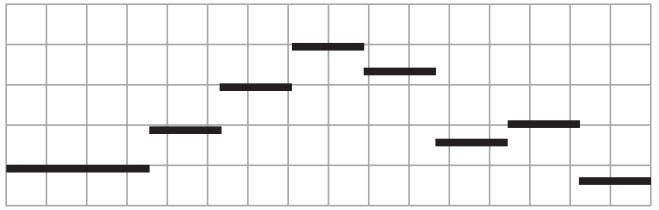
The lengths of the stages relative to each other are initially calibrated to be equal to each other (all the stages have the same duration) when the Curve toggle is set to the middle position. Due to the analog nature of the Facehugger, the calibration may shift for various reasons, for example:

- The speed gets into the audio frequency range and higher.
  - As the speed increases, the reset portion of the circuit begins to dominate in latency and causes the last stage to lengthen.
- Temperature variation
- Time
- Gremlins

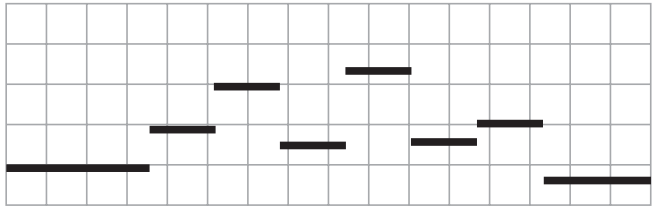
Here is how to recalibrate your Facehugger should it become necessary:



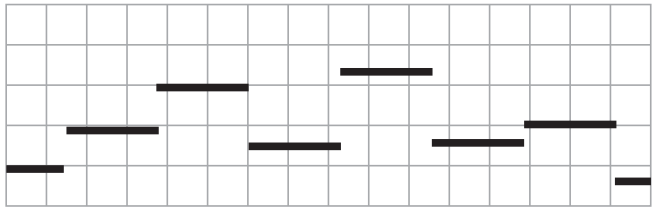
Evenly adjusted



Stage 1 long



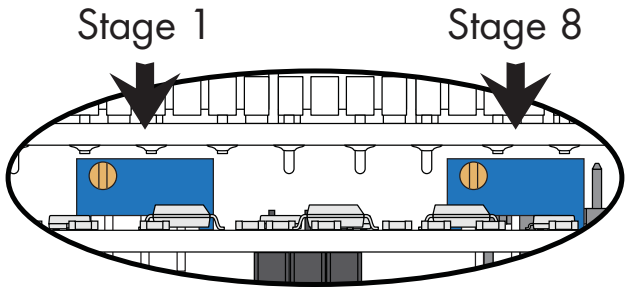
Stage 1 and 8 long



Stage 1 and 8 short

Stage length adjusting (cont.)

Trimmer potentiometers RTS1 and RTS8 adjust the lengths of the output stages: RTS1 controls the length of stage 1, RTS8 controls the length of Stage 8.



This is easiest to do if you have an oscilloscope to view the output. You can technically also do it by ear, but that takes a bit longer, from my experience.

Another method is to measure the top (pad 8/+) voltage, divide by eight and multiply by stage number to derive the voltages for each stage:

v = pad8

pad7 = pad8 x 7/8 = v x 7/8

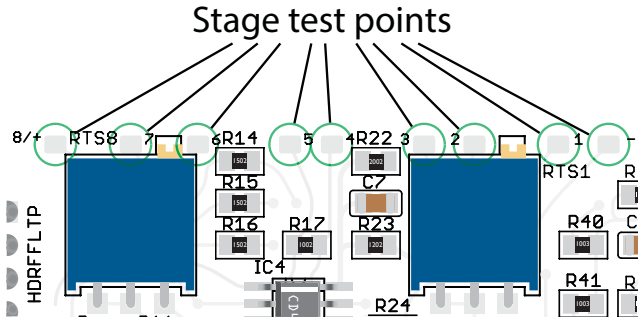
pad6 = v x 6/8

and so on...

Use the test points to observe the voltages as you make the adjustments. Stage "-" is the bottom, and should measure zero. If not, subtract its value from pad 8 before dividing:

v = pad8 – pad1.

pad 7 = (pad8 – pad1) x 7/8 = *v* x 7/8



As alluded to before, however, perfection ain't gonna happen. Facehugger is not a clock-driven, synchronized device, it's signal-driven.

Facehugger block diagram