"Sitting at the round metal coffee table, on the deck of my beach house, I cannot tell whether it will come inland or stay hovering over the waters; I know it is a storm cloud because of the lightning, extending to space and salt, exchanging energies."

Ieaskul F. Mobenthey --- *Swoop* Manual --- Page 1

The *Swoop* module is essentially a bound/bounce control voltage processor, in the lineage of such multi-purpose tools as the "Serge Dual Universal Slope Generator". Like that module, *Swoop* responds to:

• input voltages that set target boundaries in voltage

• control voltages for angle of both the down and up segments of the waveform

• trigger voltages that perform a single "swoop" up or down.

It differs, however from the DUSG, in that it truly is a free-running oscillator, that will bounce between any given boundaries; bounds take a more correlative role in deciding the frequency. In fact, it is already running when given zero input, bouncing infinitesimally at a very high resultant frequency. This oscillator was conceived as a sort of triangular answer to the operation of filters, that take an input and resonate damp. Damping, in *Swoop*, results in an ever higher and higher frequency, thus fulfilling the bounds/bounce concept. Inspired by the idea of polarity, there are actually two triggers, for upward swoop and downward swoop, or even both ways consecutively.

Swoop is an 8HP Eurorack module that runs on +12 and -12 volts. Attach power connector positive to "+" and negative to "-". Failure to follow proper power polarity will result in instant destruction of unit.

Looking at the front panel of *Swoop*, note that inputs are marked by copper fill. The two bounds inputs each control both upper and lower boundaries. They do this by rectification. With no connections, they are set at zero, and thus as mentioned already, oscillator is squeezed into this infinitesimal space. If one connection is
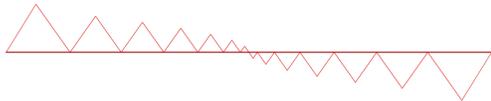
made, the oscillator will bounce up and down between zero and the voltage, whether it be positive or negative. Likewise, with two inputs, the oscillator bounces between whichever is highest and whichever is lowest.

Up in the bounds zone, there is a range switch. When in middle position, *Swoop* runs at a standard audio rate. Pointing downward is a low audio rate, and pointing upwards is a definite CV, lowest rate.

The triangular output is the position of the wave, and square is the mechanical signal to go up or down, a good way to hear its frequency.

The bounce section is a very standard sort of analog math block. There is a bounce "basis" knob for both down and up slopes, to set the base rate. Above it, there is an attenuverter, marked by a bow-tie, to process the modulation inputs, at top. In addition to the down and up inputs, there is a "both" input,
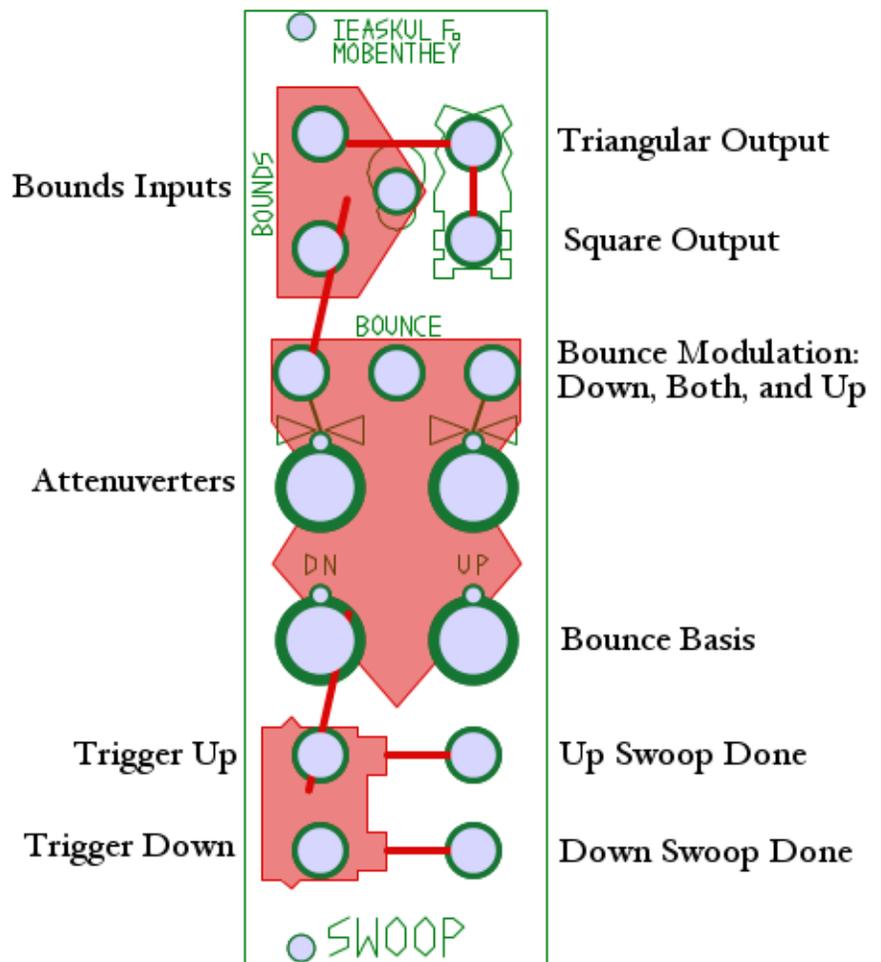
"Sitting at the round metal coffee table, on the deck of my beach house,
I cannot tell whether it will come inland or stay hovering over the waters;
I know it is a storm cloud because of the lightning, extending
to space and salt, exchanging energies."

Ieaskul F. Mobenthey --- *Swoop* Manual --- Page 2

in the middle for convenience. That input feeds into both the down and up inputs, which have separate attenuverters. An attenuverter works like this: at noon the modulations are nulled out, they have no effect; clockwise from there they increase in intensity, with positive input meaning "more"; to the counter-clockwise direction, modulations increase as well, but with negative input meaning "more". This knob is essential to controlling how much, and in which direction, your modulations apply.

At the very bottom of the module is the triggered swoop section. As mentioned above, there are two swoops, negative-going and positive going. The trigger responds to a positive transition, crossing one diode drop above ground (one volt). This should work with pretty much any gate signal but also analog signals larger than two volts peak to peak. The outputs here are gate signals, from zero to twelve volts, indicating that the swoop has finished its duty cycle. These can be used to trigger the other swoop to make a chain of different polarity swoops, or connected in "x" formation, generating a continuous oscillation. To make a recirculating positive swoop, you can connect its output to its input, likewise for a recirculating negative swoop. The analog swoop voltage is "read" at the triangular output, up top. They cause the signal to travel once to the eight (or negative eight) volt mark.

Lineage of Swoop

I guess I first worked on swoop in the Dogslit, and the associated Conrad Papers. These were all paper circuits, originally designed for controlling the light boxes of Dan Conrad. Each swoop was like a unit cell, creating this triangular event, and then triggering another swoop; montages of loops and other topographies could be constructed from multiple swoops. These then controlled a Dogvoice (to be released by Ieaskul in Eurorack form); the multiple overlapping events resulted in unpredictable growls and barks, like how a dog wants to think nomadically.

There were swoops and anti-swoops, meaning some went up and some went down. Physically the circuits were separate. Around this time I realized how much they paralleled the DUSG, and how Serge musicians will use that module as well to create montages of cellular units that trigger each other.

Fast forward about six years, at the development of the Shnth device, a digital synthesizer. It used an internal language to process, and indeed emulate analog functions. The opcode "swoop" was designed to enable granular synthesis with triangular events, but it also happened to emulate the primitive piezo flex gesture: press down and release result in an upward swoop immediately followed by a downward swoop. It worked this way in the default "dirac" mode, or using signed math. But it also had an "arab" mode wherein swoop operated with just one positive going triangle. It so happened that signed math generated this unique "double bubble" sound from the positive swoop to negative swoop crossing zero.

There was one more purposing of the term "swoop" in code, as a synthetic response function for enemy chubes in the game "Mikey Walker". Chubes respond to punches by wobbling and then falling if punched in time with their resonance: that's the aim of the game. So I was seeking to provide another type of response than the standard (but totally wonderful) resonant filter. In a separate chapter I can describe how I used Anthony Braxton's tri-centric logics as a brainstorm for the idea of using a triangle wave as a resonant filter. Essentially it needs to have an input, not a modulation input but an input as like a filter, and it needs to damp energy down by oscillating it. Of course, when a triangle wave gets bigger and smaller its frequency changes, according to the bounds/bounce concept; this helped enrich gameplay with a new resonant strategy for enemies.

So in Ieaskul's swoop, we have several threads of analog, emulation, and conceptual coming together in one module. There is biphasic support, swoop and anti-swoop are not separated anymore; but along with signed math also comes unsigned math. There is the emulation of gestural flex, in Newton's own "equal and opposite reactions". And there is the possibility of a "resonant triangle" and what happens when it is damped: Ieaskul's masthead, the paradox wave.