SYNC

Documentation

1. Introduction

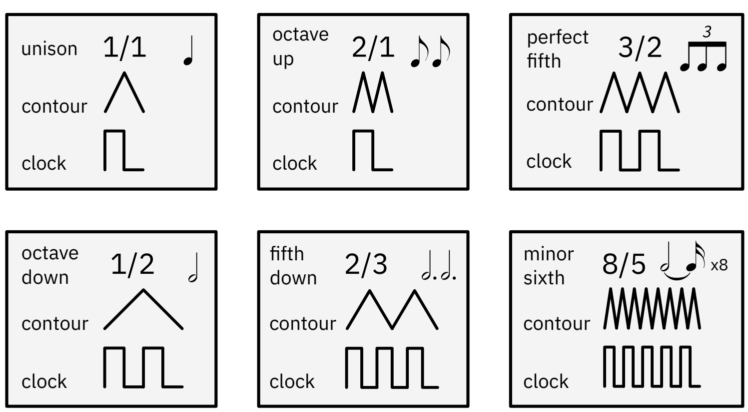
A pair of signals with frequencies related by a ratio are useful in musical synthesis. For the sub-audio signals, the ratio represent a rhythmic relationship; for audio, it represents a tonal relationship.

If this doesn't immediately click, imagine an instrument playing quarter notes at 120 BPM. It plays two notes per second. An accompanying instrument plays eighth notes, or four notes per second.

Thinking of each new note as one "cycle", the note frequency is 2 notes per second for the first instrument and 4 notes per second for the accompaniment, also understood as a 2/1 frequency ratio. At any tempo, the instrument playing eighth notes will play twice as many notes as the instrument playing quarter notes, and the 2/1 ratio remains.

Extending this, imagine two oscillators with the same frequency relationship, but with one at 440Hz and one at 880Hz. The quarter note/eighth note relationship is no longer relevant, but rather is replaced by a similarly universal musical concept, the octave, which is just a doubling of frequency. This is not the only musically relevant interval generated by a ratio of simple integers; the perfect fifth (3/2), perfect fourth (4/3) and any just-intoned interval are all derived in this manner.

Below is a figure of various ratios, their relationships as intervals and rhythmic values, and what a contour at that frequency would look like relative to a timing clock.



The effective space of rhythms and intervals is far from a discrete set of integer ratios (drummers and equal tempered tuning protest), but these idealized frequency relationships can be a solid musical foundation.

SYNC borrows the crossfader configuration used by the META and SCANNER modules, but sets the crossfade contour timing with a clock input and a ratio.

The clock input provides an external reference frequency. It can accept a wide frequency range, allowing operation as a modulation or sound source. The maximum measurable clock time is about 59 seconds, with measurement quality gradually degrading as the clock frequency exceeds 1khz. SYNC can serve as a master clock source as well; the tempo can be tapped out on the pushbutton.

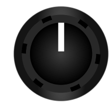
The **X** and **Y** controls pick a ratio from a "scale", arranged as a 2-dimensional grid. The selected ratio and clock input determine the frequency of the contour generator. Each scale has a particular quality, with some suited for tempo-synced modulation, others quantized pitches, and a few designed for all of the above.

The simple (integer) ratios allow the contour generator to lock to the phase of the clock input. A range of methods are available with increasing correction speed. The modes elicit unique frequency artifacts as they impose phase alignment.

A patch starts with your choice of clock input. From there, you can control ratio selection, crossfader sample and hold behavior, and contour shape. A set of auxillary outputs creates a palate of options for adding musical structure to your patch.

2. Controls and IO

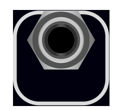
If you haven't yet, take a glance at this introduction of Via's controls, IO, and user interface.



Knobs

RATIO X and **RATIO Y** are manual controls for selecting the ratio from the scale grid, setting a position in the X and Y axis.

controls the shape of the crossfade contour. Each **WAVE** has its own smoothly-adjustable set of contour shapes. They are generally arranged in order of increasing complexity.

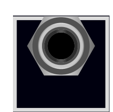


CV

RTO X CV is combined with the **RATIO X** knob to select the current ratio along the X axis.

MOD CV can be set to three destinations, *Ratio Y, Phase Offset,* or *Skew,* selected with the **MOD** parameter.

CV morphs the shape of the crossfader contour. 0 to +5V will span between the position set by the manual control and full-scale morph, and 0 to -5V between the manual control position and minimum morph.

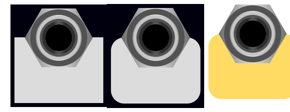


Logic Input

SYNC is the master clock input of the module. Clean square, pulse, or negative-ramping saw waveforms provide the most solid clock. Simple waveforms such as sine, triangle, and positive-ramping saw will give expected results as well, but the phase of the output will be shifted somewhat. The base frequency of complex waveforms won't be extracted; instead the calculated frequency will jump between multiple values. This can create interesting but unpredictable results.

A rising edge at the **RESYNC** input causes the phase to reset to 0 on the next rising edge.

Outputs



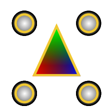
A X B The analog output derives its crossfade position between the **A** and **B** inputs from the contour generator.

ΠL is a reconfigurable logic output, selected by holding shift and tapping **S+H**. It is either high during the "attack" part of the contour and low during the "release" part, or it is high while moving towards B and low while moving towards A, tracking the changes in direction of a complex waveform.

/\ is a reconfigurable signal output, selected by holding shift and tapping **MOD**. It can be a triangle with a phase offset from the contour generator set by the  shift-function, or it can be a direct output from the contour generator.

on the expander produces a trigger when the frequency ratio changes. The length of the trigger is the gate length of the input clock. This signals a change in pitch due to a changing ratio if the contour generator is oscillating at audio rates.

LED Display



When a touch sensor is pressed, the LEDs show the mode of the corresponding parameter.

The white LEDs connected to the **A** and **B** inputs are illuminated whenever the sample and hold on that channel is holding a value. The LED connected to the main logic output is illuminated when the logic level is high, and the LED connected to the **MOD** CV input gives information about the **MOD** mode: constantly illuminated when in *Phase Offset* mode, off when in *skew* mode, and in *Y Modulation* mode it blinks each time it crosses into an adjacent row.

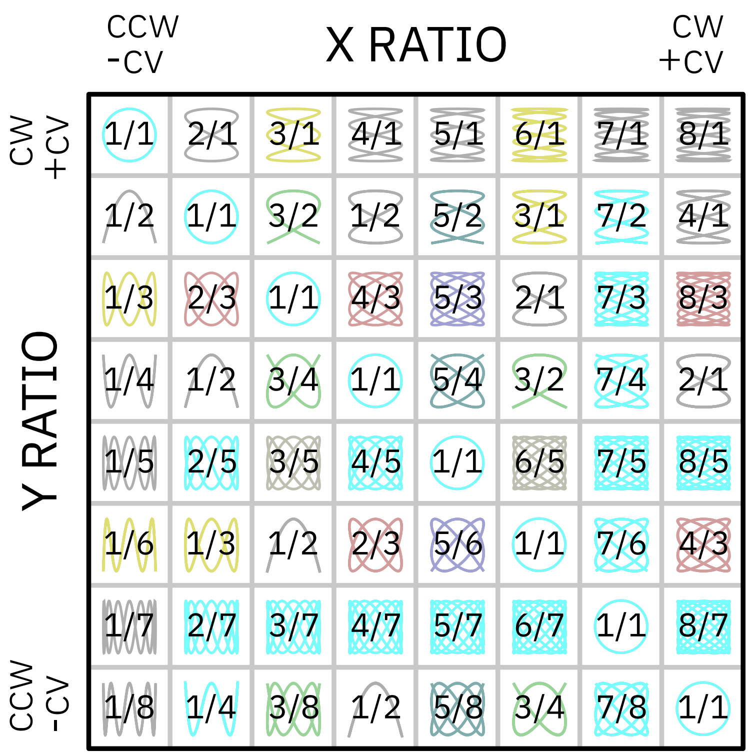
The RGB display shows a hue corresponding to the current scale. Each group has a unique base color, and each scale in the group shifts the hue displayed. The display is off when the contour generator is at A, and follows the contour in intensity as it moves closer to B.

The red/green LED above it shows the state of the analog output. It is off at 0V and becomes brighter green with positive voltages, and brighter red with negative voltages.

3. Parameters

SCALE

A **SCALE** is a two dimensional grid of frequency ratios. Ratios are selected from the grid using the **X RATIO** and **Y RATIO** controls and corresponding CVs. An example scale could be visualized as follows:

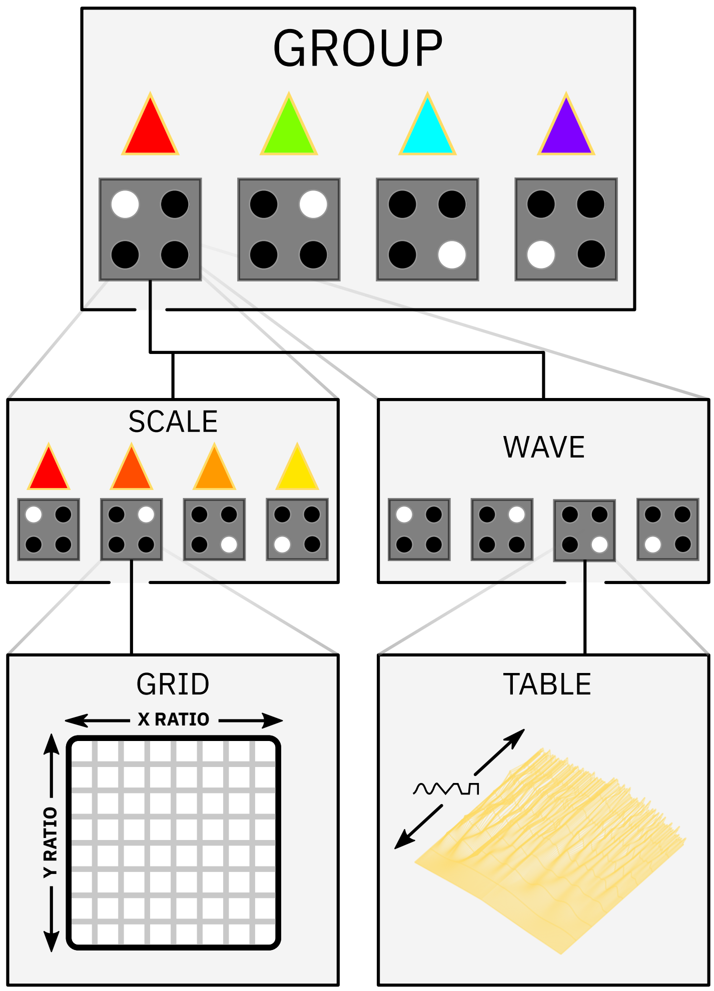


WAVE

WAVE cycles through the bank of four wavetables in the current group. A wavetable is a morphable space of waveforms, with smooth interpolation from waveform to waveform. If the **AUX WAVE** parameter is set, it will cycle through the four global waveforms, irrespective of the current group.

GROUP

GROUP selects one of four groups. Each group consists of 4 scales and 4 corresponding wavetables, with an organizing principle.



The first group is organized with an ascending numerator on the X axis (steadily increasing frequency multiplication) and an ascending denominator on the Y axis (steadily increasing frequency division). The available tables are appropriate for use as control voltages or audio signals.

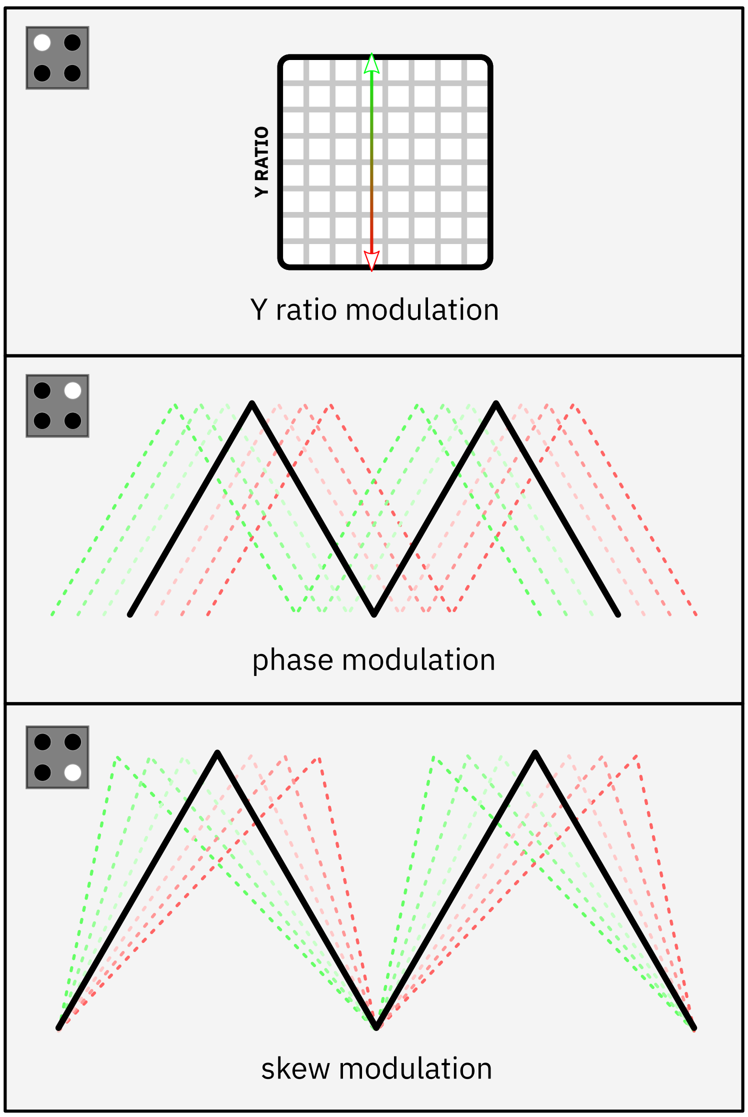
The second lays out a field of arpeggios that can be scanned across both dimensions and a set of tables for audio use.

The third is tuned for V/oct quantization with similarly audio-focused tables.

The fourth is organized for generating rhythms, and the tables tend towards envelope and LFO shapes.

MOD

MOD routes the the **MOD CV** (CV2) to one of three possible destinations: (1) the *Y* dimension in the scale grid, (2) *phase offset*, or (3) *skew* of the contour generator. *Phase offset* and *skew* accommodate input signals through the audio range, whereas *Y modulation* is best suited to lower frequency CV signals.

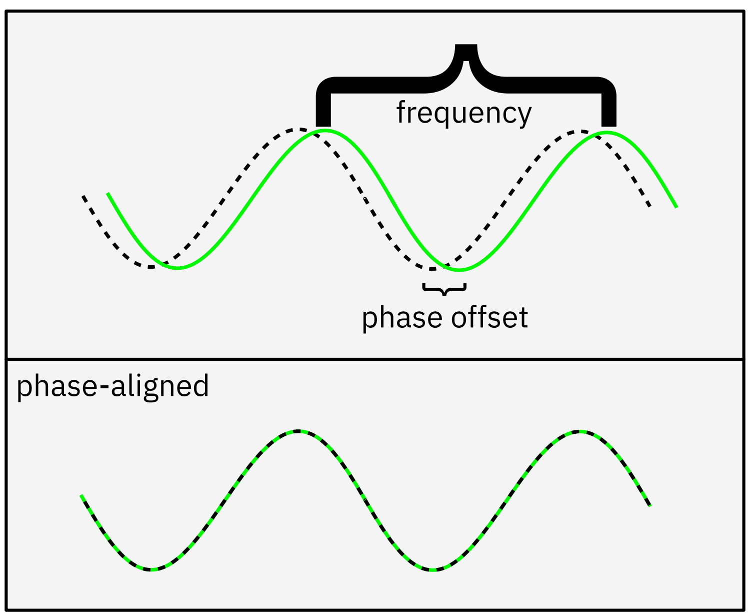


SYNC

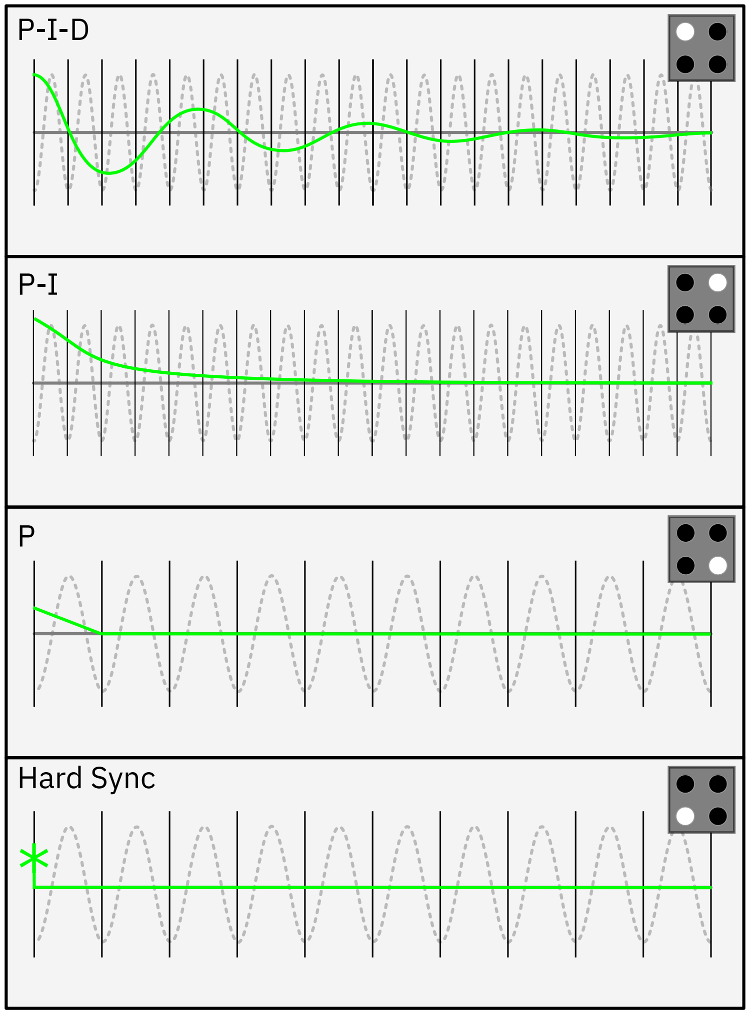
As previously mentioned, the contour generator can sync to the phase of the clock input. There are four possible sync settings: (1) *P-I-D*, (2) *P-I,* (3) *P*, and (4) *Hard Sync*. Practically, the phase correction happens faster for each mode. Hard sync provides the fastest response but imparts a discontinuity on the signal from the sudden jump in phase when it makes a correction.

The choice of ratio will influence how stable the phase locking is, and how quickly it settles: phase adjustments always occur when the clock input should coincide with a new cycle of the contour generator.

For any who are curious about where the names come from, the first three implement a phase locked loop, adjusting the frequency of the contour generator over time to align its phase with the input clock, not unlike a DJ beat matching a record. They use proportional, integral, and derivative terms (from PID control theory) to do the correction.



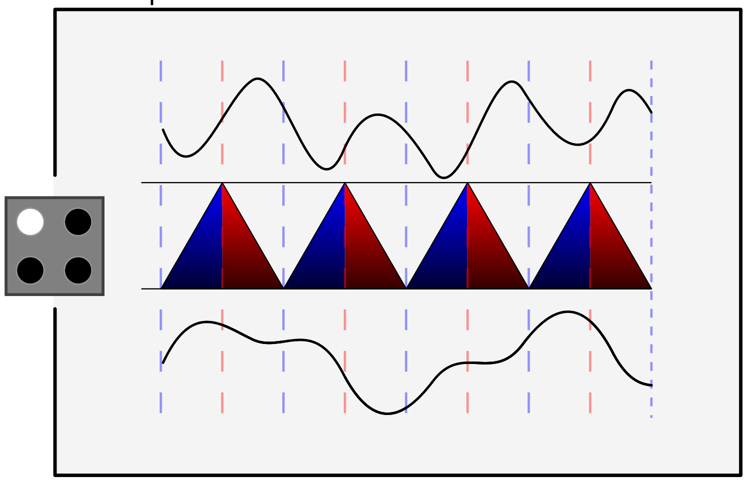
The following diagrams help describe how phase error of the contour generator (in green) changes over time relative to the input clock. The vertical lines are points where phase adjustments can occur and the dashed line represents an input clock. Depending on the given ratio, this may happen on every rising edge of the input clock, or once every N cycles.



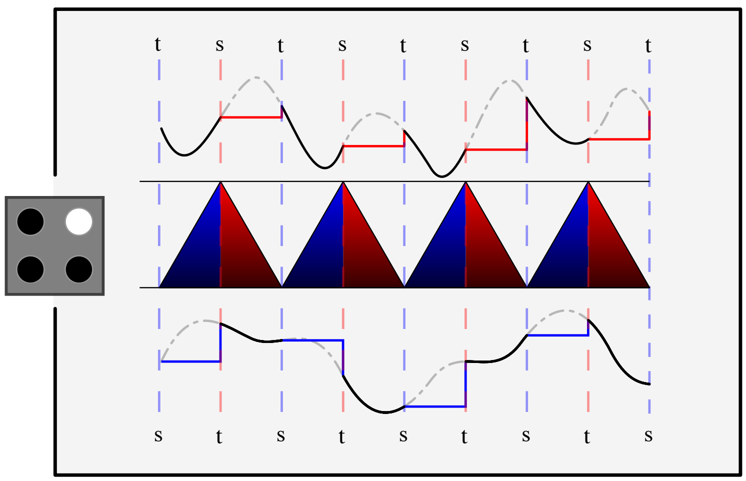
S+H

The sample and holds at the analog inputs behave identically for **A** and **B** in each of the three modes. In mode (1), they both continuously track the input. In mode (2), they sample the input when the crossfade contour reaches that channel and release the sample when the crossfade contour reaches the other channel. In mode (3), they resample the input every time the crossfade contour reaches the other channel.

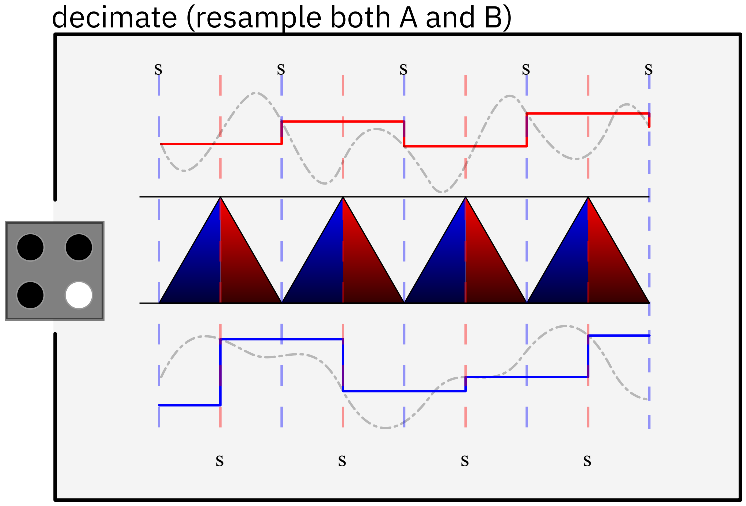
no sample and hold



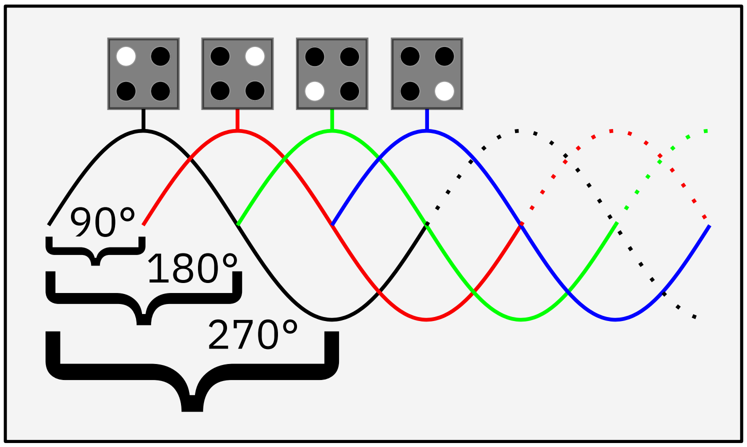
track and hold both A and B



decimate (resample both A and B)



Holding **SHIFT (GROUP)** and tapping **SYNC** sets a fixed phase offset from the input clock in 90 degree (aka quarter-cycle or quadrature) increments.



AUX WAVE

The **AUX WAVE** parameter, toggled by holding **SHIFT (GROUP)** and tapping **WAVE**, enables a "global" bank of four broadly useful wavetables that do not change with the GROUP parameter.

⎍ and ⌃

The function of the ⎍ output can be selected by holding **SHIFT (GROUP)** and tapping **S+H**.

The function of the ⌃ output is selected by holding **SHIFT(GROUP)** and tapping **MOD**.

The available options are described in the Controls and IO section.

4. Scale Directory

GROUP 1

Harmonics

The scales are arranged such that **RTO X** sets the numerator and **RTO Y** sets the denominator from a set of ascending integers. As such, a clockwise turn of **RTO X** increases frequency and a clockwise turn of **RTO Y** decreases frequency.

1. **Integers** - Integers up to 8, well suited for general purpose use.
2. **Evens** - Even integers to 16, also well suited for general use.
3. **Mult 3** - The first 8 multiples of 3, which elicits triplets and an unusual overtone/undertone series.
4. **Odds** - Odd integers through 15, enforcing odd overtones and creating Bohlen-Pierce intervals.

GROUP 2

Arpeggios

RTO X scans through a row of ratios containing harmonically related pitches. **RTO Y** selects the row from a set of complementary options.

1. **Modal Tetrads** - Each row contains an arpeggio that walks up the inversions of a seventh chord. The root note of seed chord progresses up the major scale as **RTO Y** is turned clockwise.
2. **Major to Minor** - Arranged similarly to scale 1, the seed chords move from a major mode to a minor mode as **RTO Y** is turned clockwise.
3. **Impressionist** - A clockwise turn of **RTO X** scans up one mode of a scale and then down a complementary mode. **RTO Y** guides the scales through a harmonic movement according to minimal voice leadings.
4. **Bohlen-Pierce** - Similar to scale 3, but the scales are modes of the Bohlen-Pierce scale spanning a single tri-tave. B-P scales use only odd integers, so they are complemented by a wavetable based on triangle shapes containing only odd harmonics.

GROUP 3

V/Oct Quantization

RTO X scans through a row of pitches scaled to v/oct at the CV input. **RTO Y** selects the row from a set of related scales.

1. **Modes** - Each row contains a modal scale such as major, dorian, mixolydian, etc. The rows move through the possible modes as **RTO Y** is turned clockwise.
2. **Tetrads** - Each octave of the row contains 4 pitches which are chosen to consonant equal tempered intervals . Each row is seeded with a different set of pitches.
3. **Harmonic Entropy** - A s **RTO Y** is turned clockwise, pitches are added to each octave according to a Stearn-Brocot tree. At the middle of the range, pitches are subtracted according to moving up a Farey sequence. This gives open intervals at either extreme of the control and a denser cluster of more dissonant pitches in the middle
4. **Bohlen-Pierce** - Similar to group 2, scale 4, but the scales are extended across the full v/oct span of the CV.

GROUP 4

Rhythms

he scales are arranged such that **RTO X** selects a frequency ratio representing a rhythm relative to the "tempo" provided at the clock input. **RTO Y** sets the available rhythms. Some scales enforce a phase reset after a certain number of input clock cycles.

1. **Rhythmic Harmonics** - The ratios are arranged in the fashion of group 1, but the integer set is chosen to create the frequency ratios associated with common rhythms.
2. **No Triplets** - Similar to scale 1, but without ratios containing a multiple of 3.
3. **Multiplier with Reset** - Similar to scale 1, but all multipliers are a multiples of 3.
4. **Dotted** - Similar to scale 1, but with dotted rhythms.

5. Table Directory

GROUP 1

Audio and Modulation

1. **Additive Evens** - Mimics a smooth transposition through increasing even harmonics.
2. **Ramps** - Each slope consists of an increasing number of ramps; existing ramp start and end points stay in place as new ramps are added.
3. **Bounce** - Morph through snapshots of bouncing ball simulation with increasing number of bounces.
4. **Circular** - The shape morphs from a logarithmic-like shape to an exponential-like state. Intermediate shapes have flatted regions between curve slopes.

GROUP 2

Audio

1. **Impulse Train** - Impulse waveforms with increasing overtones resembling a filter sweep without resonance.
2. **Tri to Square** - Morph from a triangle to an additive recreation of a square wave.
3. **Perlin Noise** - One dimensional slices from a spacial noise algorithm, increasing the frequency of noise per waveform.
4. **Synthesized Vowels** * - Morph through renders of a modeled vocal tract using an impulse train and a filter bank tuned to the resonances of different vowels.

GROUP 3

Audio

1. **Additive Pairs** - Emerging pairs of odd overtones.
2. **Filter Model** - A square wave through a modeled transistor-ladder lowpass filter with increasing cutoff frequency.
3. **FM** - Samples from a two-operator FM configuration with increasing modulator frequency.

4. **Sampled Train Whistles** - Morph through waveforms re-synthesized from train whistles.

GROUP 4

Modulation

1. **Symmetric expo/log** - A pair of exponential curves transform into a pair of logarithmic curves.
2. **Asymmetric expo/log** - The first half morphs from a bowed out logarithmic curve to a sharp exponential curve. The second half morphs between the same shapes in the other direction.
3. **Steps** - The slopes are stepped, morph adds steps.
4. **Sequences** - Interpolate through a set of 16-step sequences, with 8 steps per slope.

Global

Audio and Modulation

1. **Odd Triangles** - Mimics a triangle wave transposing through increasing odd frequency multipliers.
2. **Odd Sines**- Mimics a sine wave transposing through increasing odd frequency multipliers.
3. **Euclidean Ridges** - Distribute ridges across the slopes according to the euclidean algorithm and morph through underlying patterns.
4. **Skip Saw** - Impose sawtooth ridges on a triangle wave with an increasing number of ridges.

6. Presets

Each preset is geared towards an input clock frequency range. The parameter settings enable a particular type of patching strategy suited to the input clock.

Harmonic Oscillator (S+H button)

GROUP: *1*
SCALE: *Integer Ratios*
WAVE: *Additive Evens*
SYNC: *Fast PLL*
MOD: *Ratio Y*
S+H: ***Sample and Track***
AUX: *Off*
⋀ **Out:** *Triangle*
Logic Out: *Square*
Quadrature: *90 degrees*

Example patch

Arpeggiated Oscillator (SCALE button)

GROUP: *2*
SCALE: *Impressionist*
WAVE: *Impulse Train*
SYNC: *Medium PLL*
MOD: *Ratio Y*
S+H: *Off*
AUX: *Off*
⋀ **Out:** *Contour*
⊓⊔**Out:** *Delta*
Quadrature: *0 degrees*

Example patch

Bohlen-Pierce Modes (MOD button)

GROUP: *2*
SCALE: *Bohlen-Pierce*
WAVE: *Odd Triangles*
SYNC: *Slow PLL*
MOD: *Phase Modulation*
S+H: *Resample*
AUX: *On*
∧ **Out:** *Triangle*
⊓ **Out:** *Delta*
Quadrature: *90 degrees*

Example patch

Modal Quantizer (SYNC button)

GROUP: *3*
SCALE: *Modes*
WAVE: *Sampled Train Whistles*
SYNC: *Slow PLL*
MOD: *Phase Modulation*
S+H: *Off*
AUX: *Off*
∧ **Out:** *Contour*
⊓ **Out:** *Square*
Quadrature: *0 degrees*

Example patch

Sequencer (GROUP button)

GROUP: *4*
SCALE: *No Triplets*
WAVE: *Sequences*
SYNC: *Hard Sync*
MOD: *Ratio Y*
S+H: ***Resample***
AUX: *Off*
∧**Out:** *Contour*
⊓ **Out:** *Square*
Quadrature: *0 degrees*

Example patch

Tempo-synced LFO (WAVE button)

GROUP: *4*
SCALE: *Rhythms*
WAVE: *Additive Evens*
SYNC: *Fast PLL*
MOD: *Skew*
S+H: *Off*
AUX: *Off*
∧ **Out:** *Triangle*
⎍ **Out:** *Delta*
Quadrature: *0 degrees*

Example patch