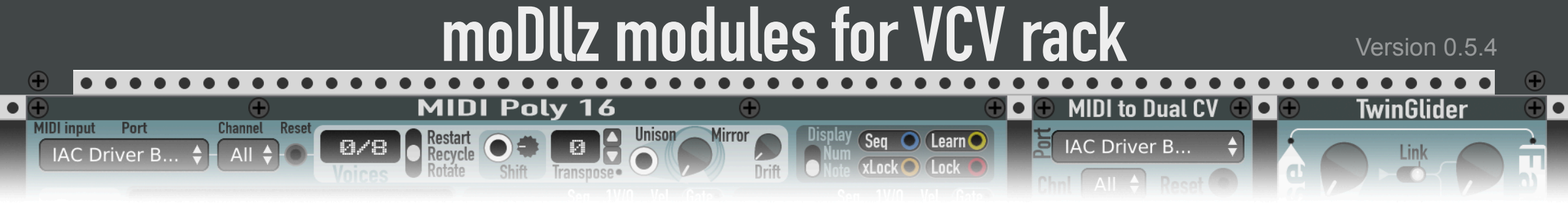
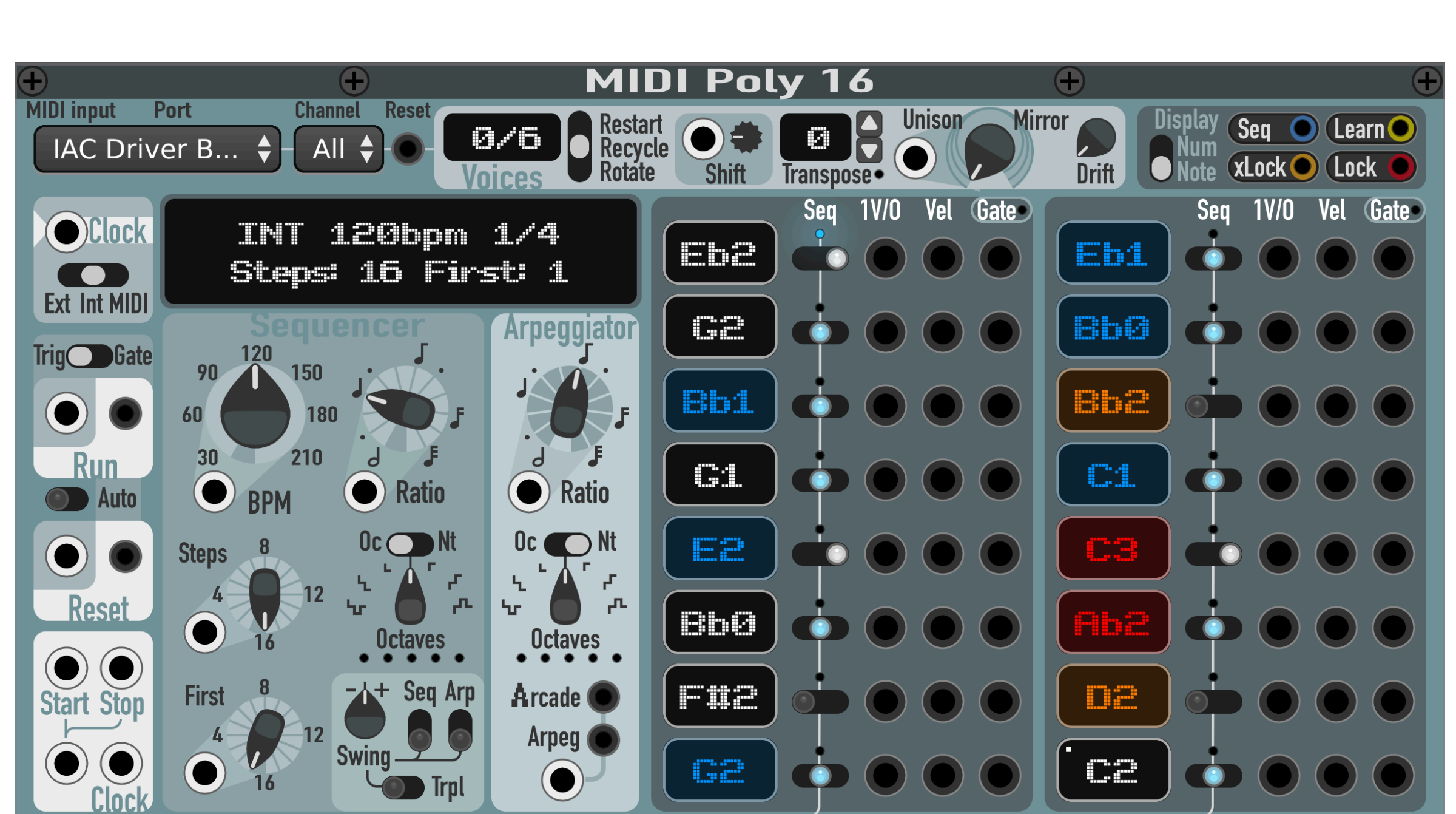


MIDI Poly 16

Poly & Mono interface + Sequencer + Arpeggiator

This module converts MIDI notes (and basic controllers data) to CV, providing up to 16 independent "voices". It also has a Mono note output (lowest, highest or last played note) with Arpeggiator and a 16 step Sequencer.



The incoming MIDI notes are dynamically assigned to 16 "Pads", with independent 1V/Oct, velocity and gate outputs each.
The Pads display the Note name (or MIDI note number), and briefly show the Velocity value when receiving a new note. (Pads can be clicked for quick preview of their output)

The Pads work in 4 different modes.

POLY The dynamic or live polyphonic mode is the default setting. The Pads just play-through the incoming notes.

SEQ On Sequencer mode, the Pads are played by the Sequencer. They ignore the incoming notes, thus reducing the number of "live" voices. The note assigned to them is memorized, and can be reassigned engaging the **Learn** function.

With Learn engaged, a clicked Pad will turn to "learning state" **Eb3** awaiting for the next MIDI note.
After receiving and saving the note, the Pad goes back to its normal state.
While on "learning state" the can also be transposed with the transpose buttons (note that the transpose display stays on its value)
To assign/unassign Sequencer mode, engage the **Seq** button and click the desired pad.

LOCK On Locked mode, the Pads are locked to the assigned note,so they play when the incoming MIDI note matches. The incoming note also plays normally through the [Poly] pads. In locked mode, notes can also be assigned with the **Learn** function.

To assign/unassign Locked mode, engage the **Lock** button and click the desired pad.

XLCK Locked Exclusive mode is similar to [Locked], but the matched note does not play through the [Poly] pads. This basically blocks that note out from the Poly section.

To assign/unassign Locked Exclusive mode, engage the **xLock** button and click the desired pad.

Poly Pads Voice assignment modes:
Restart: use first available
Recycle: repeat last used and continue to next avaliable
Rotate: use next available

Poly notes 1v/o outputs Shift +/- 48 semitones limited by Trim knob by semitone steps (Modulation is not stepped)

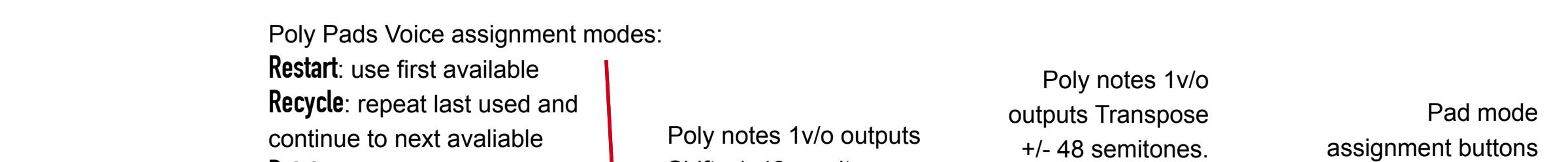
Poly notes 1v/o outputs Transpose +/- 48 semitones. Notes remain non-transposed on the Pad display

Pad mode assignment buttons (Seq Learn xLock Lock)

Poly Voices Played of total Available (depending on Pad modes)

Pad display: Note names or MIDI note number

MIDI input port / channel and Reset



Poly Bender : ~ Unison ~ Mirror
Each of the playing Poly notes is modulated up or down to match the Mono note (which is automatically selected according to the Mono output voice setting) *
Unison occurs at halfway. After that the modulation continues until reaching the mirrored note interval (with the Mono note being the axis)
If a CV 0-10v input is connected, the Knob controls the Range of modulation (0v is zero , 10v is Knob value)

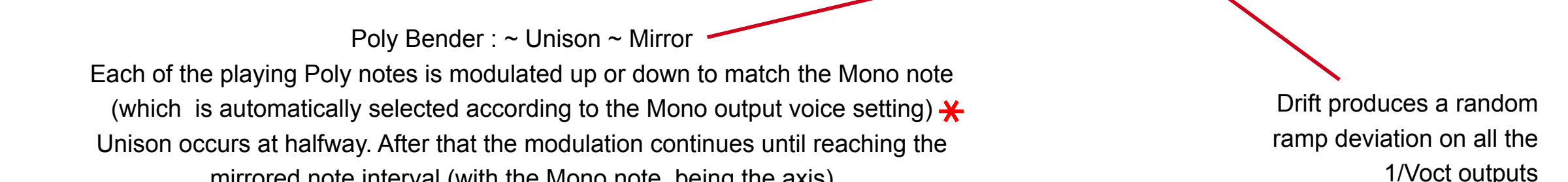
Drift produces a random ramp deviation on all the 1/Voct outputs simulating a non-stable analog source

Sequencer 1v/o outputs Transpose +/- 48 semitones

Sequencer Outputs with Gate mute preview (mute is not saved)

Mono Outputs (with Gate mute preview) and MIDI controllers

Locked Outputs (with Gate mute preview)



Sequencer 1v/o Output Shift +/- 48 semitones limited by Trim knob by semitone steps (Modulation is not stepped)

Re-Trigger on every step

*Mono output voice from Poly Pads **Upper** bound **Last** played or **Lower** bound

Hi-Res Pitch Bend +/- 5v

Modulation CC1, Channel Aftertouch and Sustain CC64

Sustain notes hold

include xLock Pads

Mono output voice from Locked Pads **Upper** bound **Last** played or **Lower** bound

Run Sequencer external Trigger/ Gate options Auto Reset to First Step on Run

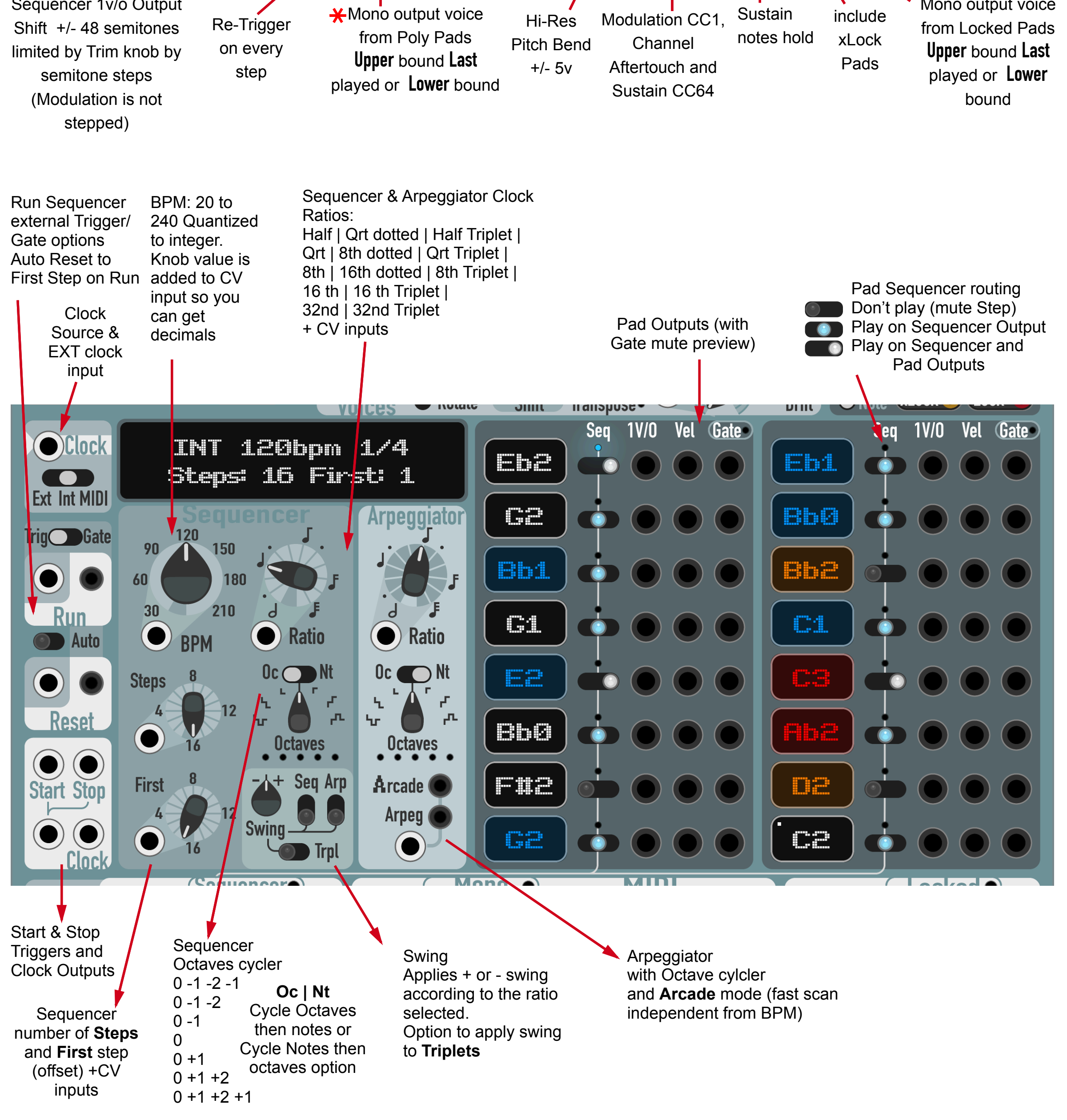
Clock Source & EXT clock input

BPM: 20 to 240 Quantized to integer. Knob value is added to CV input so you can get decimals

Sequencer & Arpeggiator Clock Ratios:
Half | Qrt dotted | Half Triplet |
Qrt | 8th dotted | Qrt Triplet |
8th | 16th dotted | 8th Triplet |
16 th | 16 th Triplet |
32nd | 32nd Triplet
+ CV inputs

Pad Sequencer routing
Don't play (mute Step)
Play on Sequencer Output
Play on Sequencer and Pad Outputs

Pad Outputs (with Gate mute preview)



Start & Stop Triggers and Clock Outputs

Sequencer number of **Steps** and **First** step (offset) +CV inputs

Sequencer Octaves cycler
0 -1 -2 -1
0 -1 -2
0 -1
0
0 +1
0 +1 +2
0 +1 +2 +1

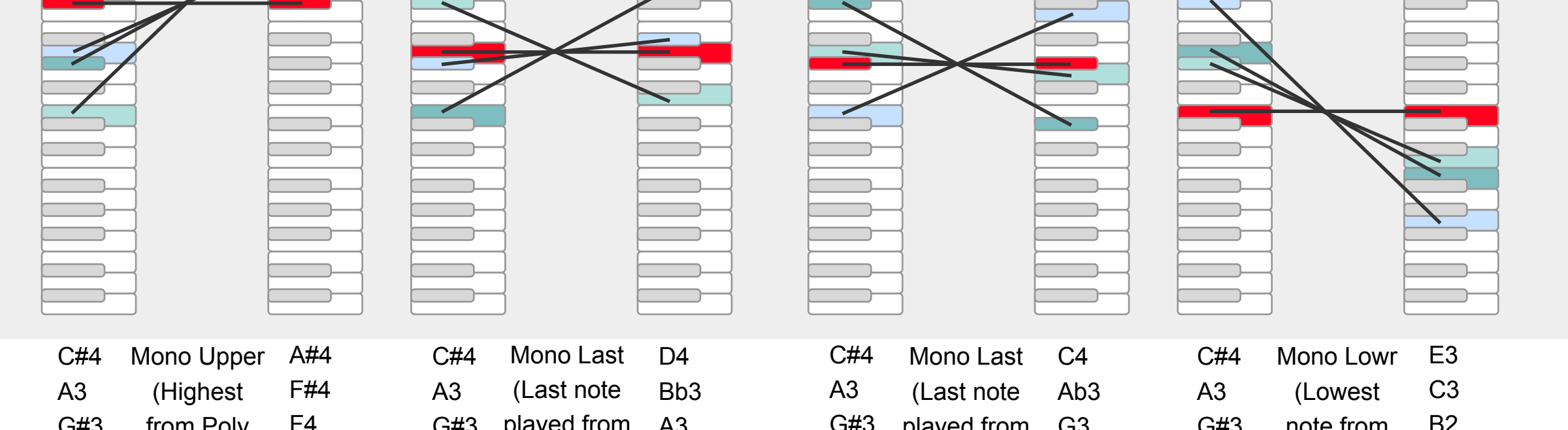
Oc | Nt
Cycle Octaves then notes or Cycle Notes then octaves option

Swing
Applies + or - swing according to the ratio selected.
Option to apply swing to **Triplets**

Arpeggiator with Octave cylcler and **Arcade** mode (fast scan independent from BPM)

Poly Bender example:

Here's a chord (E3 G#3 A3 C#4) processed with the Poly Bender, with different Mono settings. At half way all notes match the Mono note. Then they cointinue up/down until the mirrored chord



C#4 A3 G#3 E3	Mono Upper (Highest from Poly =C#4)	A#4 F#4 F4 C#4	C#4 A3 G#3 E3	Mono Last (Last note played from Poly. = A3)	D4 Bb3 A3 F3	C#4 A3 G#3 E3	Mono Last (Last note played from Poly. = G#3)	C4 Ab3 G3 Eb3	C#4 A3 G#3 E3	Mono Low (Lowest note from Poly. = E3)	E3 C3 B2 G2

More examples here:
https://www.facebook.com/dllmusic/videos/10155837031103787/

MIDI to dual CV interface

Based on : Core MIDI-To-CV

Simulates a duophonic keyboard sending 2 CVs / Vel / Gate (reTriggered) for Lower and Upper notes being played
... if only one note is played Lower and Upper CVs are the same (Useful for unison / separate OSC with optional RingMod / Oscillator sync)
There's also different options to retrigger Lower/Upper Gates independently.

MIDI input Port selection

MIDI input channel

Lowest MIDI note being played converted to CV

Latest lower note velocity played (CV)

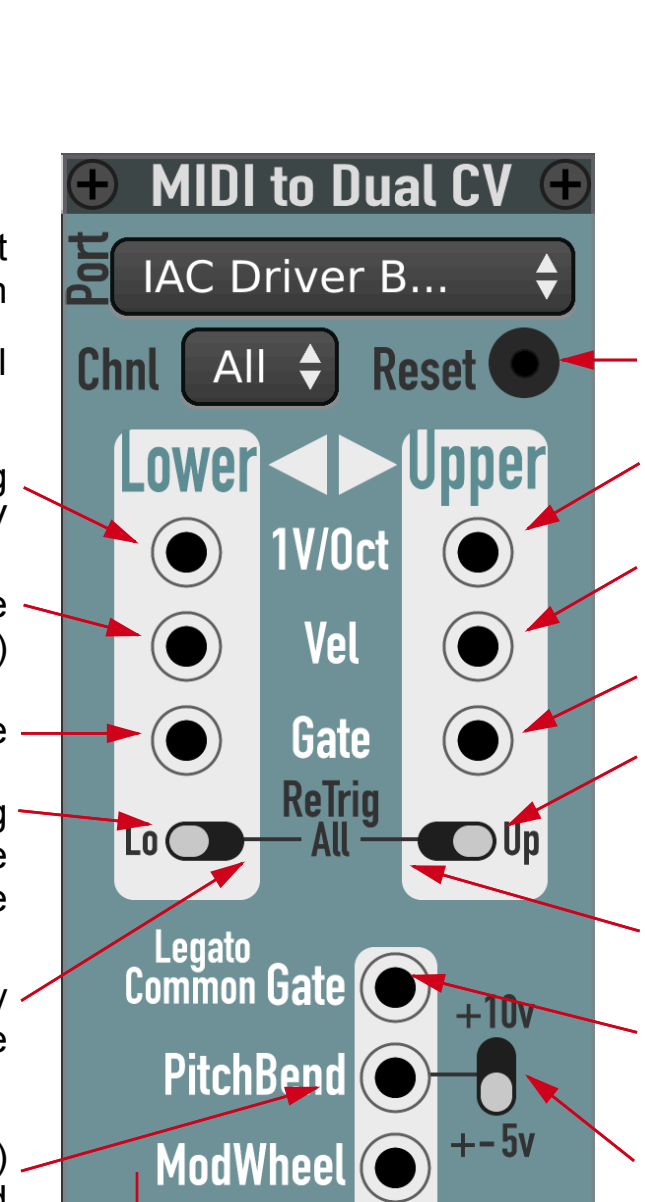
Lower note Gate

Gate retriggers if the incoming Upper note is higher than the previous one

Gate retriggers on every incoming Upper note change

Hi Res (14bit) PitchBend

Commonly used MIDI control change controllers and aftertouch



MIDI reset

Highest note being played converted to CV

Latest upper note velocity played (CV)

Upper note Gate

Gate retriggers if the incoming Upper note is higher than the previous one

Gate retriggers on every incoming Upper note change

Common Gate. No retrigger

PitchBend CV range. Defaults to +/- 5v

Gate(s) held by sustain pedal (default)

Duophonic Wiki...
https://www.sequencer.de/synth/index.php/Duophonic

TwinGlider

Based on : Befaco Slew

Twin Glider is a dual linear glide/portamento module.
It limits the rate of change of the incoming signal with independent Rise and Fall times.

Link Fall Time/Rate, CVin and Mode to Rise values

Rise Time/Rate

Rise Time/Rate CVin (delimited by knob value)

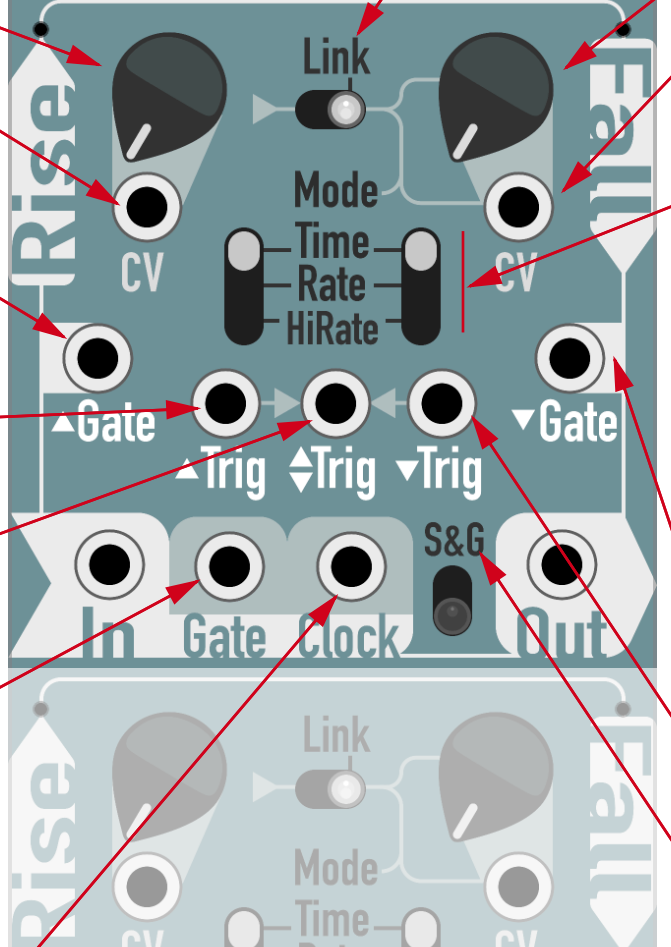
Rise Gate +10v while Fall ramp is active

Rise Trigger : Triggers when Rise ramp is done

Rise+Fall Trigger: Triggers when any ramp is done

Gate in: When connected, the Rise/Fall ramps are active only if a Gate is received. Connect note gates here to produce portamento on legato notes only

Clock input for Sample to clock event and Glide



Fall Time/Rate

Fall Time/Rate CVin (delimited by knob value)

Glide Mode:
constant **Time**: (ramp time independent from input jump)...
constant **Rate** (ramp time directly related to input jump)...
High Rate for processing audio (fast changing) signals

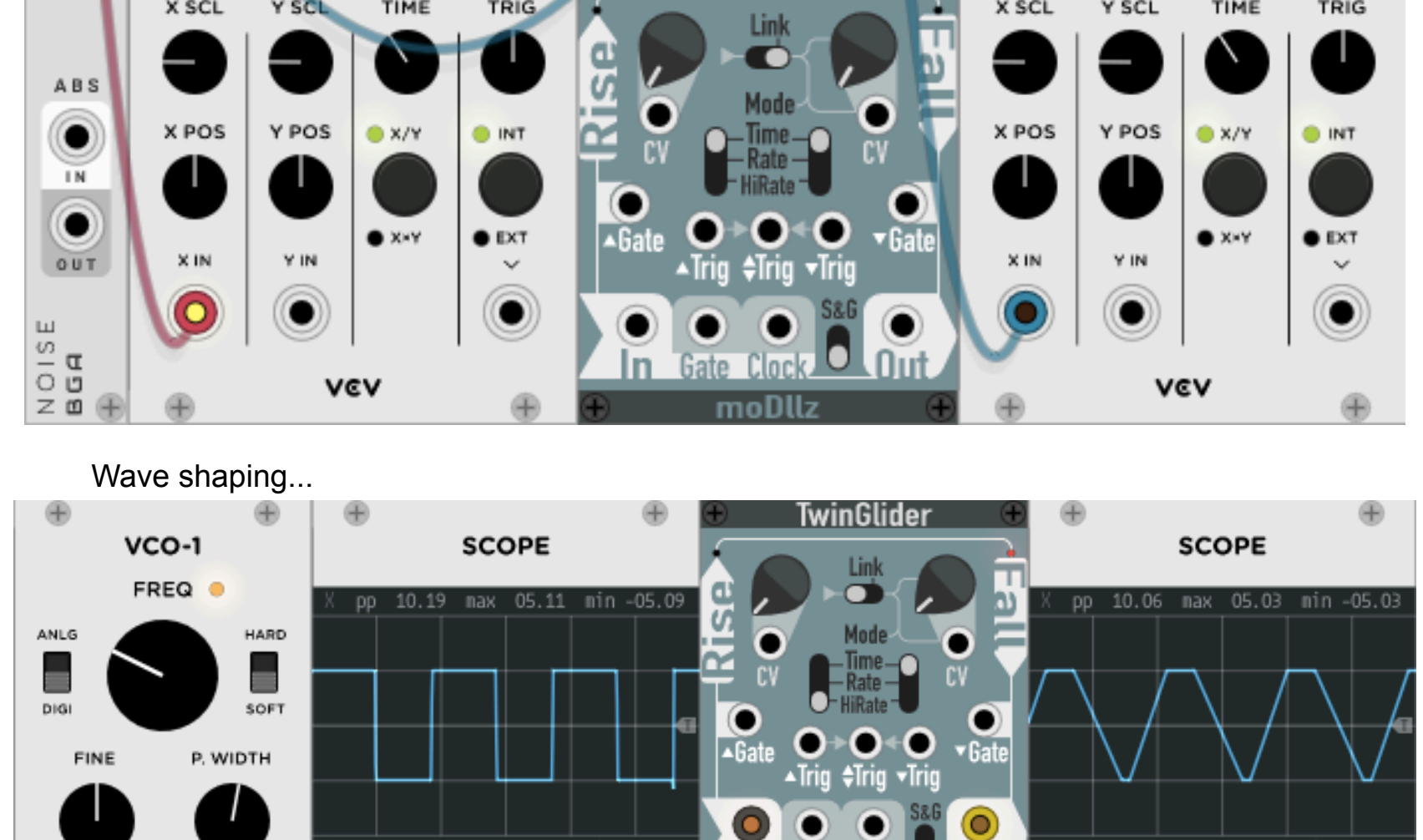
Fall Gate +10v while Fall ramp is active

Fall Trigger: Triggers when Fall ramp is done

Sample & Glide:
(Without connected Clock) The input signal is sampled only when ramp is not active (Input is disregarded until the ramp is done).
When Clock is connected the input signal is sampled on every clock, and disregarded between clocks.

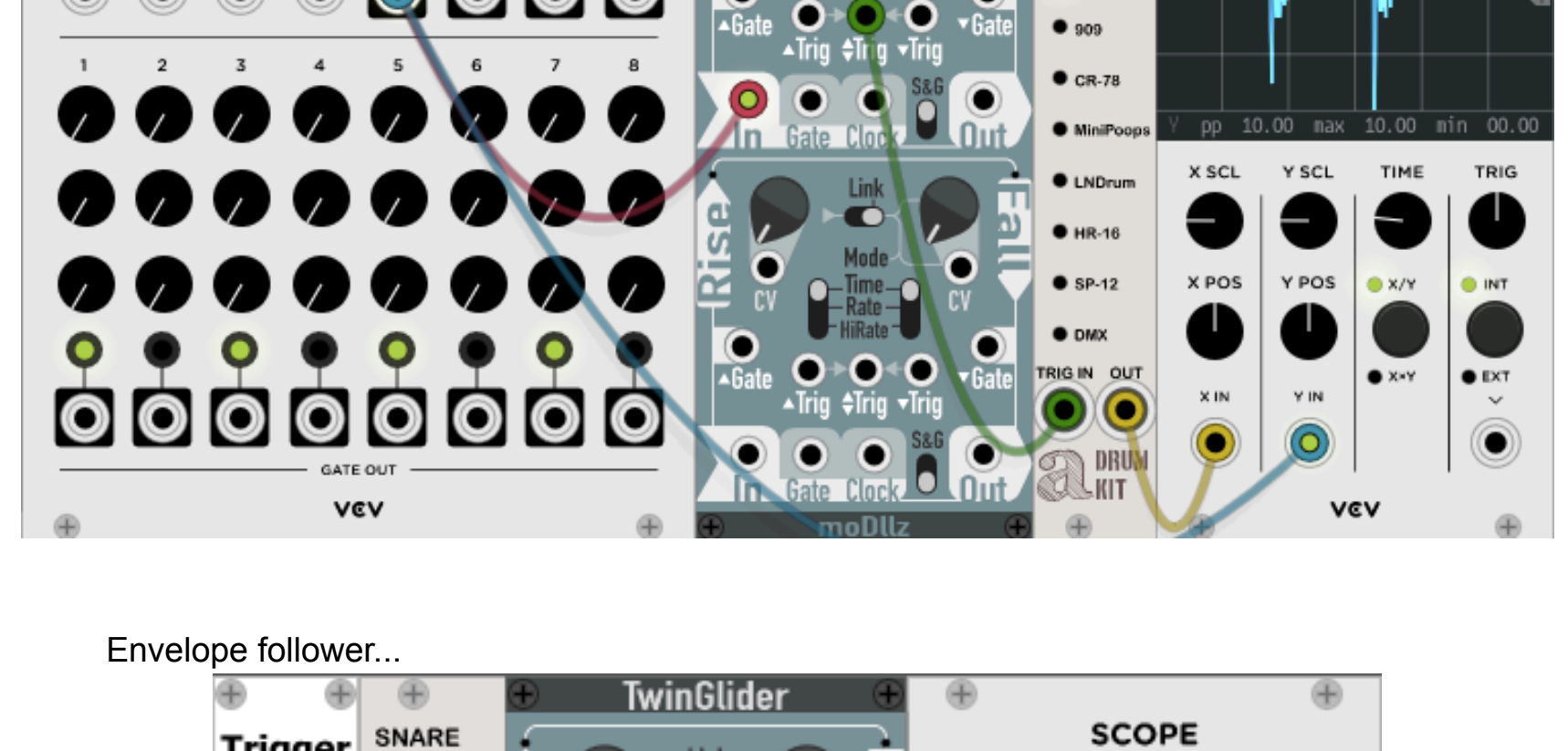
Example usages:
Note portamento (1v/oct Input > Out to VCO)
Get "slow" noise signals for random modulation (noise Input : S&G, Time mode)
Wave shaping "filter" (signal Input: HiRate mode)
Generate Trigger signals at the beginning and ending of a Gate (Gate to Main Input (not the Gate input) zero times (ramps inactive) > rise+fall Trig Out)
Generate Gates from Trigger (Trigger Input : with Rise (or Fall) time > Rise (or Fall) Gate Out)
Envelope Follower (Audio Input : with Fall time

"Slow" noise...

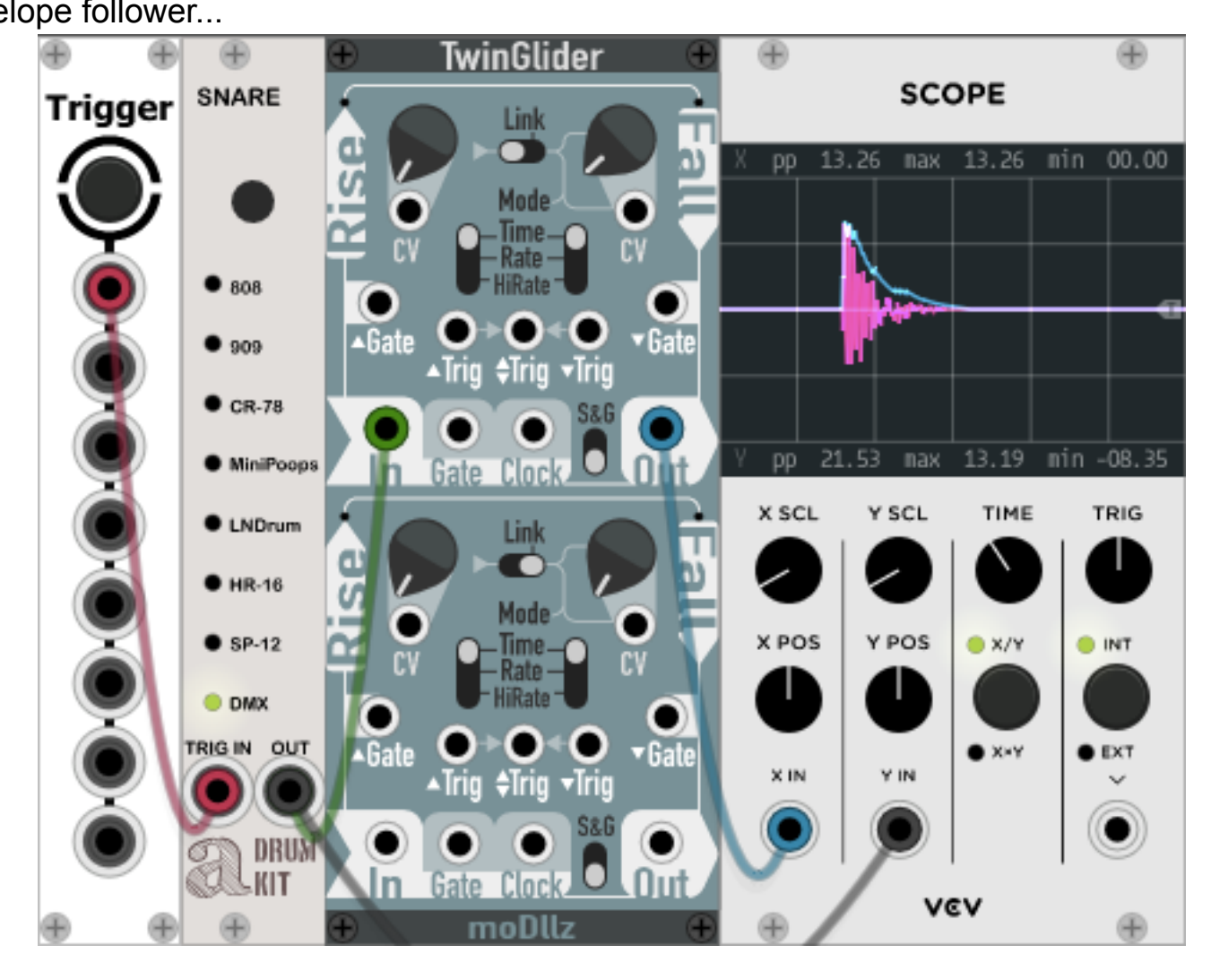


Wave shaping...

Gate up/down triggers...



Envelope follower...



moDllz Example patches...
https://github.com/dllmusic/VCV_moDllz/blob/master/patches/moDllzVCVpatches.zip?raw=true