

# Modular Reference

## MIDI

- MVC A/B - MIDI Voice Control A/B
- Key Split
- Key Zone
- MIDI Clock
- Clock Divider
- MIDI to Trigger
- MIDI Channel Filter
- MIDI Monitor

## Gate

- Gate2Sync
- Gate Switcher

## OSC

- Frequency & Pitchmodulation
- SyncMaster/SyncSlave
- Multi OSC (SyncM/SyncS)
- Pulse OSC (SyncS)
- Saw down/up (SyncS)
- Sine OSC
- Uknow OSC (SyncM/SyncS)
- Morphing Pulse
- Morphing Saw
- Spectral OSC
- Fm Operator
- Tube Resonator
- Noise Generator
- BPF Noise
- Sample Oscillator
- Sample Pool

## Envelopes

- ADSR
- ADSR B
- AD (& Mod) Vintage
- AHD (& Mod) Vintage
- ADSR (& Mod) Vintage
- Multisegment Envelopes
- Envelope Follower

## Mix & Gain

- PolyOut 1 & 2
- Linear und Exponential VCA
- Slope Mod VCA
- 6dB/12dB Gain
- Volume Attenuator
- Micro Mixer
- Mix 2/4/8
- Static Crossfade
- Static Pan
- Crossfade Modulator
- Pan Modulator
- Amplitude Modulator
- On/Off Switch
- 1x4 Switch (Gain)
- 4x1 Switch (Gain)
- Peak Meter

## Filter

- 6dB Highpass
- 6dB Lowpass
- 12dB Multimode Filter
- 18dB Lowpass
- 24dB Lowpass Filter
- 24dB Highpass Filter
- 24dB Lowpass Filter V
- 24dB Lowpass Filter R
- 24dB Highpass Filter R
- 24dB Bandpass Filter R
- Multimode Filter A
- Multimode Filter B
- Uknow Filter
- Combfilter A/B
- Vocal Filter
- 12dB Lowshelf EQ
- Parametric EQ
- 12dB Highshelf EQ
- Free Filter Bank A/B

## LFO

- Single/Poly und LFOs
- Multi LFO A
- Multi LFO B
- MW LFO
- Pulse LFO
- Saw Down LFO
- Saw Up LFO
- Sinus LFO
- Square LFO
- Triangle LFO
- Random Signal Generator
- Sample & Hold

## Modifiers

- Pitch Modifier A
- Pitch Modifier B
- Pitch Modifier C
- Constant Value
- Constant Value bipolar
- Constant Freq
- Curve Table
- Linear Scale
- Frequency Divider
- Frequency Multiply
- Pitch Quantizer
- Xmod & Feedback Connector

## Drum

- Drum Synth
- Drum Oscillator
- Percussion Oscillator
- Drum Voice Control
- Mute Adder 2
- Hihat Source
- SampleDrum Oscillator
- MDS 8

## Effects

- Compressor
- Limiter
- Delay
- Tempo Delay
- Chorus
- Flanger
- Tempo Flanger
- Phaser
- Tempo Phaser
- Distortion
- Bit Quantizer
- Decimator
- Ringmodulator
- Mono/Stereo Insert

## Sequencer Modules

- Gate SEQ1 / SEQ2
- Ctrl SEQ1 / SEQ2
- Ctrl SEQ B1 / SEQ B2
- Pitch SEQ
- Pattern SEQ
- PS32
- GateOr
- Start/Stop

# MIDI

The modules in this group process MIDI data. Some modules process existing MIDI signals, while others supply MIDI control signals to other synthesis modules.

## MVC A/B - MIDI Voice Control A/B

The MVC modules are the most important in the MIDI group. They are required in every synth patch. They make the connection between your MIDI keyboard or sequencer and your synth patch, interpreting MIDI information such as note number, note velocity, pitch bend and aftertouch (with influence from settings on the MVC module itself) and translating this info into frequency, gate, velocity and other signals which can be used to control oscillators, filters, envelopes, and so on.

There are two versions: MVC A offers portamento, while MVC B does not and is therefore "less expensive" in terms of DSP power.

## Controls

### Coarse/Fine

Transposes the pitch of the incoming **Freq Out/Sample F** note data. Coarse transposes the pitch in semitones, while Fine adjusts it in *cents* (hundredths of a semitone).



## PWR

The Pitch Wheel Range setting controls the deflection of the pitch by the incoming pitchwheel values. The range is adjustable from 0 to 24 semitones. The pitchwheel has no effect, of course, if the value here is set to zero.

## Porta/Glis

When Portamento or Glissando is switched on, the pitch will either glide (Portamento), or progress in a stepwise sequence (Glissando), from one note to the next over a fixed time period (adjustable - see next section).

You can adjust this parameter to Off, Portamento (P), Glissando (G), fingered Portamento (fP) or fingered Glissando (fG). Note that Portamento/Glissando is only effective when playing in a legato style.

## Time

Used in conjunction with the Port/Gliss option, above. This sets the amount of time to glide or gliss from one note to the next.

## Curves

Opens the **Velocity/Aftertouch Curve** dialog with additional parameters.

### Velocity/Aftertouch Curve

The illustration shows the 128 possible Velocity/Aftertouch values.

The associated parameters Curve, Sensitivity, and Offset all contribute to the way the MVC adapts the Velocity/Aftertouch values to produce the desired response. The fundamental response is determined by the 7 Curves - linear (curve 1), fixed (curve 3), exponential (curve 4) or logarithmic (curve 6). Sensitivity adjusts the depth of the curve (i.e. the degree of response) and Offset adds or subtracts a fixed amount from the response curve.

For more information, see the section 'Curve Table', also in this chapter.



## Connections

### MIDI In

Connect the MIDI signal from a **MIDI Source** module, or another modular module that has a MIDI out port to this input.

### Freq Out

This jack supplies frequency values for oscillators, pitch modifiers etc. You can also connect an LFO's Freq In input here.

### Sample Freq

This output is used for frequency control of the SampleOscillator.

Do not confuse this setting with the sample rate setting. A sample rate designates the bandwidth of a selected sample. The Sample Freq connection is for pitch information used for sample playback.

### Note

Supplies the MIDI Note value. Use this to connect to a Key Follow modulation input or other modulation input (such as **EnvelopeTmod**).

### Vel

Provides the MIDI velocity value after having been 'massaged' by the velocity curve. This output can be connected to several different inputs to provide modulation based on keyboard dynamics. It is commonly used to influence envelope timings and levels, and filter cutoff modulations.

### AT

Provides MIDI Aftertouch values after having been processed by the aftertouch curve. Attach any modulation input to this output.

### Gate

Connect envelope generators, LFOs, or other modules that require a gate signal to this output. Some envelope generators use Envelope Synchronization (Esync, see next section) which must be connected in conjunction with Gate in order for the voice envelopes to function correctly. You can connect several envelope generators in parallel to the Gate output.

### Esync

This is an input used to receive Esync (Envelope Synchronization) messages. Esync signals are fed back from the envelope generators and report the current state of the envelope to the MVC. This is necessary for precise voice control. Only if Esync and Gate are both connected to their appropriate ins and outs will the envelope generators operate error-free. To use more than one envelope, you must also include the Esync Adder module.

**Important:** If you use the gate signal with no envelope generator present in the patch, you must first connect Gate with Esync (i.e. bridge the connections in the MVC). Otherwise, no gate signal will be sent.

## Key Split

This module takes the MIDI data and splits it according to note number into two keyboard zones defined by a split point. Only MIDI data falling within the low or high split zones are transmitted through the respective Low/High MIDI outputs. Each zone has its own Transpose parameter. Typically you would connect two MVCs to the outputs of the Key Split module to control two separate sounds.

### Controls

#### Low Key

This value sets the lowest note (as a note number) for the keyboard zone. The highest note for the zone is determined by the Split value.

#### Split

This note number separates the high and low keyboard zones.

#### High Key

The note number that defines the upper boundary of the high keyboard range. The lower boundary is determined by the Split value.



#### Transpose

Allows for independent transposition values for the Low and High keyboard ranges, adjustable to +/-64 semitones. Note that the transposition will succeed only if the result lies within both the respective low/high key range and the permissible range of MIDI note numbers (0-127).

### Connections

#### MIDI In

Input for the MIDI signal.

#### MIDI Low

Outputs the low split zone portion of the MIDI signal (after transposition).

#### MIDI High

Outputs the high split zone portion of the MIDI signal (after transposition).

## Key Zone

This module reads the incoming MIDI data and re-transmits only the portion that lies within an adjusted keyboard zone. A Transpose setting allows notes within the zone to be transposed. Typically you would insert this module before an MVC to limit the input of the MVC to a specific key range on a keyboard.

### Controls

#### Low Key

This specifies the *low* key of the range as a MIDI note number.

#### High Key

Specifies the *upper* limit of the zone as a MIDI note number.



#### Transpose

You can transpose the notes in a key zone by up to +/- 64 semitones. The transposed notes must lie within the Low/High key range, and must remain within the allowable MIDI range of 0-127 or they will not play.

### Connections

#### MIDI In

Input for the MIDI signal.

#### MIDI Out

The resulting MIDI output (limited to the Key Zone range defined by the settings).

## MIDI Clock

This module converts internal or external MIDI clock messages into clock pulses for use by the Step Sequencer, or to control LFO frequencies, etc. A MIDI Clock signal consists of 24 pulses per beat. The Clock Signals and the resulting frequencies are output at a 1:1 ratio, i.e. 24 pulses per beat. Adjusting the clock frequency has some implications with regard to the Clock Divider (see next module) and the Frequency Divider (see the Modifier section). So that different devices within the SCOPE Fusion Platform can be synchronized with a Modular patch, the MIDI Clock has a MIDI Out which can be connected to the MIDI destination.

### Controls

#### Internal/External

Switches the MIDI Clock between Internal and External modes. When the indicator is lit, the module is in Internal mode.

#### BPM

Sets the clock tempo BPM (Beats Per Minute). The first field is the whole number value, and the second is the fractional value in 1/100ths of a beat per minute.



### Connections

#### MIDI In

Input to receive the MIDI signal from the Modular MIDI Source. Of course, this connection is only required in External mode.

In order to use MIDI In to control the Clock, a MIDI signal containing MIDI Clock information must be connected to the Modular's MIDI In within the Project Window.

#### Clock

Outputs the Clock signal of either the internal or external MIDI clock. Connect this to the Clock inputs of the Clock Divider or Sequencer, or to the Gate In of an envelope generator.

#### Start/Stop

Sends MIDI Start and Stop synchronization messages to the Clock Divider. Connect the Clock Divider's input of the same name to this output.

#### Freq

Sends the clock signal as a frequency. You can connect Freq directly to a Divider (see Modifiers) to provide a signal to connect directly to an LFO's Freq In.

#### MIDI Out

Provides a MIDI signal with MIDI Clock information so you can synchronize other CreamWare devices to the Modular's clock.



## Clock Divider

The Clock module always outputs 24 pulses per beat, but this is not always the number required in your patch, or portion of a patch. Often you will want to divide (or multiply) this number to produce different note length values. In addition to the Clock input, the Clock Divider provides two Clock outputs which supply subdivisions or multiples of the signal.

### Controls

#### Clock Divide

Each of the two outputs has an associated Textfader to adjust its multiplication or division factor.

**Example:** You have a MIDI Clock, a Clock Divider, and two sequencer modules. The first sequencer is to produce quarter notes, and the second, eighth-note triplets. Connect Out 1 to the first sequencer, and set it to 24/24. Likewise, connect Out 2 to sequencer 2, but adjust the ratio to 24/8. Out 1 will send quarter notes ( $24/24 = 1$  step per beat) and Out 2 will transmit eight-note triplets ( $24/8 = 3$  steps per beat).



$24/96 = 1/1$  note

$24/48 = 1/2$  note

$24/24 = 1/4$  note

$24/16 = 1/4$  triplets

$24/12 = 1/8$  note

$24/8 = 1/8$  triplets

$24/6 = 1/16$  note

$24/4 = 1/16$  triplets

$24/3 = 1/32$  note

### Connections

#### Clock

Connect the Clock output of a MIDI Clock or other module here.

#### Start/Stop

Connect this with the output of the same name on the MIDI Clock module. The MIDI Clock uses this connection to enable synchronous control. When this signal is received, Out 1 and Out 2 of the Clock Divider are resynchronized.

#### Out 1

Clock Divide output 1.

#### Out 2

Clock Divide output 2.



## MIDI to Trigger

This module converts detailed MIDI Note On events to a more generic trigger, or Clock signal. MIDI to Trigger can be used, for example, to trigger a step sequencer whenever you play a note on the keyboard. MIDI to Trigger distinguishes between Single and Legato modes.

In Single mode, each new note on event produces a trigger signal.

In Legato mode, notes played before the previous note is released are ignored. Therefore, there must be a small space between two notes for the second to produce a trigger.

### Controls

#### Single

Enables Single mode. The button is lit when enabled.

#### Legato

Enables Legato mode. The button is lit when enabled.



### Connections

#### MIDI In

Connect this to a MIDI source.

If you want the module to respond only to a specific MIDI channel, connect a MIDI Channel Filter and set it to pass MIDI data only from the desired channel.

#### Trig

Outputs the trigger/clock signal.

## MIDI Channel Filter

The MIDI Channel Filter permits only those MIDI events which are on the selected channel to pass through. This permits you to divide the common incoming MIDI stream according to channel, so that specific portions of a patch respond only to messages received on a specific channel. Multiple MIDI Channel Filters can be used within one patch to apply different MIDI channels within one patch.



### Connections

#### In

Connect, for example, the MIDI input of the patch here.

#### Out

Delivers the filtered MIDI stream, containing only the messages on the specified channel.

## MIDI Monitor

The MIDI Monitor shows the incoming MIDI-messages and allows therefor to see quickly, if a MIDI signal is present and on which channel it's coming in. Additionally you get information about the note number and the velocity.



### Connections

#### MIDI In

Connect, for example, the MIDI input of the patch here.

# Gate

This group contains modules that process, change, and/or produce Gate signals. Some modules supply processed or refreshed gate signals, while others produce control signals for the Modular's synthesis modules.

## Gate2Sync

This simple module converts incoming Gate signals into audio impulses which can be used for oscillator synchronization and restarting.



### Controls

#### on/off

Switches gate to synchronization signal conversion on or off. When the button is lit, Gate2Sync is switched on.

### Connections

#### Gate

Input for the gate signal.

#### Sync

Output for the synchronization signal.

## Gate Switcher

This module takes an incoming gate signal and routes it one of two outputs depending on the incoming Vel or Note signal from an MVC, and the range values adjusted for each output. Therefore, a gate signal can be routed dependent on either the MIDI note, or the velocity.

### Controls

#### Min/Max1

Sets the range within which the incoming gate signal will be routed to the first output.

#### Min/Max2

Sets the range within which the incoming gate signal will be routed to the second output.

#### Link

Enable this option to make Max1 and Min2 adjustable simultaneously in parallel. This makes it easier to separate the two ranges accurately.



### Connections

#### Gate

Input for the gate signal.

#### Val

Input for the Note or Vel signal of an MVC.

#### Gate 1

Gate output for the Min/Max1 range.

#### Gate 2

Gate output for the Min/Max2 range.

# OSC

The Modular provides you with many varieties of oscillator. In this section you will find many typical and familiar oscillators, as well as several interesting implementations of new concepts.

In general, the more functionality an oscillator offers, the more hungry it will be for processing resources. Using smaller, more basic oscillators when possible will help optimize DSP usage. Always try to use smallest oscillator possible in any particular section of a Modular patch.

**Tip: Start with full featured oscillators, and then replace them later with smaller versions that supply only the functionality you need.**

This module group also includes a selection of noise generators.

## Frequency & Pitchmodulation

Without a frequency signal at the Frequency In connection, an oscillator will produce no sound, as it has no indication of what frequency to generate. One easy and common way to provide a frequency value is to connect Freq Out of the MVC to the Freq In connection of an oscillator. This patch enables the oscillator's frequency to follow a keyboard.

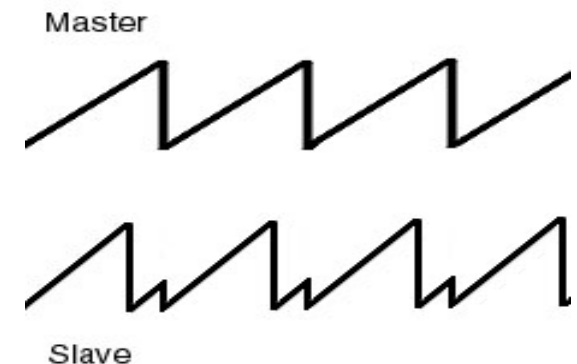
For pitch modulation, use one of the Pitch Modifiers (see under 'Modifiers'). Connect the modifier to the MVC (as you would an oscillator) and then connect the respective oscillators to the Freq Out jack of the modifier. You can connect any of several modulation sources to the pitch modifier.

If you require only a fixed frequency, use the Constant Pitch modifier in the modifier group.

## SyncMaster/SyncSlave

Some oscillators are provided in SyncMaster or SyncSlave variants. The following applies to oscillator synchronization in general:

The SyncMaster supplies a signal that adjusts the waveform of the SyncSlave to restart with each new SyncMaster cycle (see diagram). With pitch modulation, only the tone quality, but not the frequency of the slaves changes.



Normally you would connect the Sync Out of a SyncMaster to the Sync In of the slave(s). Then use a pitch modifier (see under Modifiers) to modulate the pitch of the slave(s) using any modulation signal.

## Multi OSC (SyncM/SyncS)

The classic analog oscillator with 5 waveforms: Sine, Triangle, Saw up, Saw down, and Pulse. The pulse width is controllable either manually, or by a modulation input. This oscillator is available as either a SyncMaster or SyncSlave.



### Controls

#### Coarse/Fine

Controls the pitch of the oscillator. Coarse adjusts the pitch in semitones; Fine adjusts it in cents (1/100 of a semitone).

#### Waveform

Allows you to choose from the following waveforms: Sine, Triangle, Saw up, Saw down, and Pulse.



#### PW

When the pulse waveform is selected, use PW to manually adjust the 'duty cycle' of the wave. This adjusts the shape of the pulse rectangle.

#### PwmA

Adjusts the degree to which the incoming signal at the Pwm input influences the pulse width modulation.

### Connections

#### Freq In

Input for the frequency signal.

#### Pwm

Connect a modulation signal to control the pulse width here.

#### Out

The oscillator's output signal.

#### Sync Out (Only in SyncMaster version)

Provides a synchronization signal for a SyncSlave oscillator. Connect this output to the Sync In jack of the slave oscillator.

#### Sync In (Only in SyncSlave version)

Input for the synchronization signal of a SyncMaster oscillator. Connect the Sync out of a SyncMaster here.

## Pulse OSC (SyncS)

Oscillator to produce a pulse wave with variable pulse width. Available as SyncSlave.

### Controls

#### Coarse/Fine

Controls the pitch of the oscillator. Coarse adjusts the pitch in semitones; Fine adjusts it in cents (1/100 of a semitone).

#### PW

Control to adjust the pulse width manually.

#### PwmA

Adjusts the degree to which the incoming signal at the Pwm input influences the pulse width modulation.



### Connections

#### Freq In

Input for the frequency signal

#### Pwm

Connect a modulation signal to control the pulse width here.

#### Out

The oscillator's audio output.

#### Sync In (SyncSlave only)

Input for the synchronization signal of a SyncMaster oscillator. Connect the Sync out of a SyncMaster here.

## Saw down/up (SyncS)

This simple oscillator provides descending and ascending sawtooth waveforms. Available as SyncSlave.

### Controls

#### Coarse/Fine

Controls the pitch of the oscillator. Coarse adjusts the pitch in semitones; Fine adjusts it in cents (1/100 of a semitone).



### Connections

#### Freq In

Input for the frequency signal.

#### Out

The oscillator's audio output.

#### Sync In (SyncSlave only)

Input for the synchronization signal of a SyncMaster oscillator. Connect the Sync out of a SyncMaster here.



## Sine OSC

The Sine Oscillator is provided in three versions: normal, SyncSlave, and Partial. The SyncSlave version is used for oscillator synchronization. The Partial version is specifically designed for use in additive synthesis patches, since this oscillator can be tuned to produce pitches in the harmonic series.

### Controls

#### Coarse/Fine

Controls the pitch of the oscillator. Coarse adjusts the pitch in semitones; Fine adjusts it in cents (1/100 of a semitone).

#### Partial (only in the SineOSC Partial)

This is the factor by which the basic frequency at Freq In is multiplied. By adjusting this value to integer (whole number) values, harmonics are produced: A factor of 1.000x produces the basic frequency, or first harmonic. A factor of 2.000x produces the octave above the basic frequency, or the second harmonic, etc.

Since using the rotary control to adjust this value is a little clumsy, it is better to simply enter the desired harmonic number in the text field directly. For example, enter <3> + <Enter> to adjust the oscillator to the third harmonic.



### Connections

#### Freq In

Input for the frequency signal.

#### Out

The oscillator's audio output.

#### Sync In

Input for the synchronization signal of a SyncMaster oscillator. Connect the Sync out of a SyncMaster here.

## Uknow OSC (SyncM/SyncS)

This oscillator actually consists of three oscillators: a pulse oscillator with variable pulse width, a sawtooth oscillator, and a square wave sub-oscillator. The volume of each oscillator can be independently adjusted. The phases of the pulse and sawtooth oscillators can be adjusted manually, or controlled by a modulation signal. This oscillator is provided in both SyncMaster and SyncSlave versions.

### Controls

#### Coarse/Fine

Controls the pitch of the oscillator. Coarse adjusts the pitch in semitones, Fine adjusts it in cents (1/100 of a semitone).

#### PW

Control to adjust the pulse width manually.

#### PwmA

Adjusts the degree to which the incoming signal at the Pwm input influences the pulse width modulation.



#### Ext

Switches between manual control and Modulation control of the phase of the pulse and sawtooth waves. With manual control (the button is not lit) the phases of the pulse and sawtooth waves are controlled directly using the PP and SP controls respectively. When set to Modulation (the

External button is lit) the PP and SP controls adjust the intensity of the phase modulation of the pulse and sawtooth waves.

#### PP

Controls the phase of the pulse wave. It's effect is either direct, when in manual mode, or indirect by controlling the intensity of the incoming modulation signal in Modulation mode.

#### SP

Controls the phase of the sawtooth wave. It's effect is either direct, when in manual mode, or indirect by controlling the intensity of the incoming modulation signal in Modulation mode.

#### P

Volume of the pulse wave.

#### S

Volume of the sawtooth wave.

#### Sub

Volume of the sub-oscillator.

## Connections

### Freq In

Input for the frequency signal.

### Pwm

Connect a modulation signal to control the pulse width here.

### PPm

Input for a modulation signal to control the phase of the pulse wave. The signal at this input has an effect only when Man/Mod is set to Mod (button is lit).

### SPm

Input for a modulation signal to control the phase of the sawtooth wave. The signal at this input has an effect only when Man/Mod is set to Mod (button is lit).

### Out

Audio output of the oscillator.

### Sync Out (SyncMaster only)

Provides a synchronization signal for a SyncSlave oscillator. Connect this output to the Sync In jack of the slave oscillator.

### Sync In (SyncSlave only)

Input for the synchronization signal of a SyncMaster oscillator. Connect the Sync out of a SyncMaster here.

## Morphing Pulse

This oscillator can produce a continuous waveform change (morph) from sine to pulse. The morph factor and the pulse width can each be controlled manually, or by a modulation signal.

### Controls

#### Coarse/Fine

Controls the pitch of the oscillator. Coarse adjusts the pitch in semitones; Fine adjusts it in cents (1/100 of a semitone).

#### PW

Control to adjust the pulse width manually.

#### PwmA

Adjusts the degree to which the incoming signal at the Pwm input influences the pulse width modulation.

#### WF

Here you set the morph factor which controls the relationship of the sine wave to the pulse wave.



#### WfmA

Adjusts the intensity of the modulation signal connected to the Wfm input. This signal modulates the waveshape/morph factor.

### Connections

#### Freq In

Input for the frequency signal.

#### Pwm

Connect a modulation signal to control the pulse width here.

#### Wfm

Input for a modulation signal to control the waveshape/morph factor.

#### Out

The oscillator's audio output.

## Morphing Saw

This oscillator can produce a continuous waveform change (morph) from sine to saw. The morph factor can be controlled manually, or by a modulation signal.

### Controls

#### Coarse/Fine

Controls the pitch of the oscillator. Coarse adjusts the pitch in semitones; Fine adjusts it in cents (1/100 of a semitone).

#### WF

Here you set the morph factor which controls the relationship of the sine wave to the pulse wave.



#### WfmA

Adjusts the intensity of the modulation signal connected to the Wfm input. This signal modulates the waveshape/morph factor.

### Connections

#### Freq In

Input for the frequency signal.

#### Wfm

Input for a modulation signal to control the waveshape/morph factor.

#### Out

The oscillator's audio output.

## Spectral OSC

Like the morphing oscillators, this oscillator produces a smooth transformation from one waveform to another. However, it is a little different. Although both are 'waveshaping', and produce dynamic spectra by increasing overtones for a brighter tone, this oscillator can also switch between sawtooth and pulse waveforms.

### Controls

#### Coarse/Fine

Controls the pitch of the oscillator. Coarse adjusts the pitch in semitones; Fine adjusts it in cents (1/100 of a semitone).

#### Saw/Pulse

Switches between the sawtooth and pulse waveforms.

#### WF

Here you set the morph factor which controls the relationship of the sine wave to the saw/pulse wave.



#### WfmA

Adjusts the intensity of the modulation signal connected to the Wfm input. This signal modulates the waveshape/morph factor.

### Connections

#### Freq In

Input for the frequency signal.

#### Wfm

Input for a modulation signal to control the waveshape/morph factor.

#### Out

The oscillator's audio output.

## Fm Operator

Use this module to create patches using FM synthesis. The FM Operator can be used either as a modulator or a carrier. In an FM patch, modulators determine the tone, and the carrier (the FM Operator at the end of the chain) determines the amplitude. The carrier waveform is a sine. There is a separate input for connecting envelope generators.

### Controls

#### Coarse/Fine

In an FM Operator, the Coarse/Fine controls are used not only to determine the pitch of the output, but also the spectrum (e.g. frequency modulation). Coarse changes the frequency in integer steps, resulting in simple spectra with recognizable pitch. Changes to the Fine setting result in less recognizable pitch, and more complex spectra reminiscent of bells or ring modulation. In order to use the Coarse/Fine controls, Fixed must not be enabled, and the blue LED beside the controls must be lit.

#### Detune

Allows you to detune, or fine tune the operator. Range is +/-20 cents.



#### Fixed

Fixed permits you to adjust the operator's frequency to a fixed value. You can create some particularly interesting effects using Fixed.

**When you use a fixed frequency, the Coarse/Fine controls are disabled. This is indicated when the blue LED next to Fixed is lit.**

#### Phase

Use this control to adjust the start phase of the carrier wave. Usually this results in small spectral changes, and will work only if retrigger (Ret) is enabled.

#### Ret

This switch determines whether the signal will run continuously, or be restarted at its initial phase setting each time a new note is played. Retrigger is enabled when the button is lit.

#### FmA1/2

Controls the depth of frequency modulation from the respective modulation inputs.



## Connections

### Freq In

Input for the frequency signal.

### Gate In

Input for a Gate signal (possible from an MVC). Without a signal connected here, the Phase and Ret controls will have no effect.

### EG

Input for an envelope signal. Controls the amplitude of the operator.

### Fm 1

Input for a modulator for frequency modulation.

### Fm 2

Input for a modulator for frequency modulation.

### Out

The operator's audio output. This can be connected directly to a modulation input to create a feedback loop.

## Tube Resonator

This module is actually based on a comb filter, but was conceived to perform as an oscillator. Comb filters are particularly effective when imitating sounds such as a flute (from which the module got its name, since a flute is little more than a 'tube'). To produce a flute sound, you would start by sending some noise to the resonator. The frequency is controlled, as usual, by the signal at the Freq In input. Other modulation inputs provide signals to control the spectrum.

### Controls

#### Coarse/Fine

Controls the pitch of the resonator. Coarse adjusts the pitch in semitones; Fine adjusts it in cents (1/100 of a semitone).

#### Res

Controls the strength of the comb filter effect. The higher the resonance, the clearer the pitch becomes.

#### ResMod

Adjusts the degree of influence of modulation on the resonance.



#### Damp

The resonance of the filter is created using feedback. By adjusting the Damp parameter, you control the depth of the feedback loop. This provides an additional way to control the behavior of the comb filter.

#### DampMod

Controls the intensity of a modulation signal over Damp. You can use this, for example, to apply an envelope to the Damp parameter, controlling the amount of audible noise to imitate the blowing sound of a wind instrument.

### Connections

#### In

Input for an audio signal.

#### Freq In

Input for a frequency signal.

#### Rmod

Input for a modulation signal (Resonance).

#### Dmod

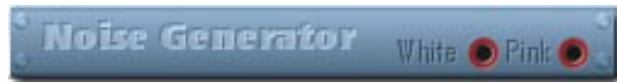
Input for a modulation signal (Damp).

#### Out

The resonator's audio output.

## Noise Generator

Noise has many applications in synthesis - as a tone color, a modulation signal, in sample-and-hold circuits - to name just a few. This module simultaneously delivers both pink and white noise at the respective outputs.



### Connections

#### White

White noise output.

#### Pink

Pink noise output.

## BPF Noise

Produces white noise that has been filtered through a bandpass filter with adjustable cutoff. The filtered and unfiltered signal is available at separate outputs.



### Controls

#### Cutoff

Adjusts the cutoff frequency of the bandpass filter.

### Connections

#### White

Output for the unfiltered white noise.

#### BPF

Output for signal after it has passed through the bandpass filter.

## Sample Oscillator

The Sample Oscillator lets you use any desired Akai (.s) or Wave (.wav) sample files as a sound source in your patch. Samples are loaded via simple drag & drop from the File Browser onto the sample-drop field. Once loaded, a sample can be played via the keyboard with normal semitone transpose per note number or, via the Fixed option, with the same pitch on all keys. The latter option is especially useful with sound effects samples. If other modules through which the sample is played, such as filters, are connected in such a way as to be dependent upon note number, then it is possible to vary a parameter (such as filter cutoff) in response to keyboard position, even though sample pitch stays fixed.

### Controls

#### Coarse/Fine

Adjusts the coarse and fine tuning of the sample oscillator in semitones and cents (1/100th of a semitone) respectively.

#### PMod

The Sample Oscillator does not receive a frequency control signal in the same sense as a standard oscillator. Instead, it receives an offset based on the Coarse and Fine Tuning settings and the pitch change introduced by Pitch Bend messages. Thus, a separate pitch modulation input is provided for connection to an LFO or similar modulation source.

Sample Drop Field



#### Low

Sets the lowest note number at which the sample is triggered.

#### Center

Sets the note number at which the sample plays at its original pitch.

#### High

Sets the highest note number at which the sample is triggered.

#### Fixed Tune

Activating this option causes sample pitch to be the same on all keys which trigger it. The Coarse and Fine Tune controls can still be used to dial in a pitch other than the original sample pitch.

**Keep in mind that transposition of a sample changes its playback speed – this is especially noticeable with sound effects samples. Transposing a sample downward slows it down, while transposing it upward speeds it up.**

#### Load

Click on this button if you want to load a sample. This button opens the standard dialog for loading files.

## Connections

### Gate

Triggers the sample. Typically connected to the Gate output of the MVC module for triggering from a MIDI keyboard or sequencer.

### Note

Presents note number information to the Sample Oscillator. Typically connected to the Note output of the MVC module.

### Smpl F

For connection to the output of the same name on the MVC module. The Sample Oscillator must be connected this MVC output instead of the Freq output normally used with standard oscillators.

### PMod

For connection to a pitch modulation source such as an LFO. Note that LFOs connected to this input should not be run at speeds higher than approximately 10Hz.

### to EG

Connect this output to the Gate input of all envelopes involved with playback of the sample. This ensures that the sample is started before the envelope receives its gate signal.

### Out

Audio output from the SampleOscillator.

## Using Samples

Once you've loaded a sample and then save a patch preset, the same sample will later be reloaded whenever you recall this preset. Keep in mind that the SampleOscillator does not store the sample itself within the preset, but merely records its name and its location. If you subsequently delete this sample, rename it or move it to a different location (or if the sample was loaded from a CD-ROM which is not inserted in the CD-ROM drive), the SampleOscillator will not be able to find it and therefore not be able to load it.

**In particular, note that when transferring a patch which uses samples to a different computer, it is also necessary to transfer the samples being used.**

To get around this problem, read the next section, Sample Pool.

## Sample Pool

The Sample Pool gives you the capability to integrate several samples into your patch in a simple manner. When you load a sample, it becomes a component of the patch. Therefore, the storage requirements of the patch rise depending on the size of the integrated samples. On the other hand, now you don't have to pay attention to where your samples are stored on your hard disk - whether they are in the correct location or not. Also, this design has the advantage of ensuring that the patch always contains the correct samples when it is transferred to another computer.

### Controls

#### Load

Press this button to load a sample, the standard dialog for loading a file appears.

#### Save Pool

Clicking on this button allows you to save the Sample Pool, in its current state, as an independent module. Later you can integrate this pool into another patch.



#### Export

Click on this button if you want to save the currently selected sample to your hard disk. This button opens the standard dialog for saving files.

#### Remove

By clicking on this button, the currently selected sample is deleted from the Sample Pool.

#### Remove All

When you click on this button all samples in the pool, regardless of the selection, are deleted.

**Samples in the Sample Pool that also reside on the hard drive will not be deleted from the hard drive by Remove or Remove All.**

**If the samples are stored only in the patch and you remove one or more from the pool, only the remaining samples will be saved when the patch is saved. The removed samples are lost when the patch is overwritten with the changes.**

# Envelopes

Envelope generators are required to produce the dynamic changes in amplitude and tone that help create the essential characteristics of an instrument. Essentially, an envelope instructs a parameter how to behave over time. Envelope generators, combined with filters and amplifiers, are essential defining elements of an instrument patch.

Most of SCOPE Fusion Platform's envelopes are suitable for any purpose - amplitude, filter, or pitch control. They feature 'Envelope Synchronization' (Esync), which makes them particularly accurate. A few were designed specifically to provide envelope control over filter or pitch only, and are named with the designation 'Modulation Envelope'. These do not include the Esync capability, and are therefore very efficient.

An envelope is triggered by a Gate On signal. It starts, and proceeds through several segments of varying pre-determined lengths. When it receives a Gate Off signal, the envelope generator jumps to the final 'release' segment.

Most of the envelopes have an adjustable Slope parameter to further define the envelope shape. Depending on the envelope generator, one of linear, logarithmic or exponential curves can be selected (see the figure beside).

**In a patch you usually use the standard envelopes that include the Esync feature. If you need an envelope generator only to control a single filter, or the pitch, then select one of the 'Modulation' envelope generators, since Esync is not required in this case. An amplifier, however, must always be controlled using an Esync-capable envelope generator.**





## ADSR

A Standard, four segment, Attack, Decay, Sustain, Release envelope generator. The slopes are adjustable, and the times and levels can be modulated simultaneously.

### Controls

#### Slope

You can adjust the curves for the attack, decay and release slopes. The curve is continuously adjustable from linear to logarithmic.

#### A

The attack time. When a gate signal is received, the attack segment starts, and continues for the maximum time duration at which point the maximum modulation level is reached.

#### D

The decay time. The envelope enters the decay segment when the attack phase completes. At this point, the modulation falls back to the sustain level. The time needed for this is the Decay time. A decay will not be heard if the sustain level is set to maximum.



#### S

The sustain level. This is the level at which the modulation signal will be held as long as the gate is open. When the gate closes, the release segment immediately follows.

#### R

The release time. When the envelope generator receives a gate-off signal, it jumps immediately from its current phase to the release segment. When the change from one segment to another takes place, the release time will be adjusted to account for the volume level at that time.

#### Tmod1

Adjusts the times of all segments of the envelope. Both the intensity and direction of the modulation effect of the signal at Tmod1 is set by this value. Negative values shorten times, and positive values lengthen them.

#### Tmod2

Adjusts the times of all segments of the envelope. Both the intensity and direction of the modulation effect of the signal at Tmod2 is set by this value. Negative values shorten times, and positive values lengthen them.

#### Lmod

Adjusts the levels of all segments of the envelope. The value here (0..max) controls the intensity of the modulation of the levels by the modulation signal.

## Connections

### Gate

Connect the gate signal from, for example, an MVC or a sequencer here.

### Esync

Connection for the Esync feedback to an MVC or sequencer module.

### Tmod1

Input for a modulation signal to control envelope times.

### Tmod2

Input for a modulation signal to control envelope times.

### Lmod

Input for a modulation signal to control envelope levels.

### Out

Output for the envelope signal.

## ADSR B

Similar to the standard ADSR envelope, but lacks the Esync output, and is therefore slightly more economical in terms of DSP power usage. This envelope should always be used in situations where it is not important that the envelope output value returns completely to zero before the envelope is restarted.



## Additional Connections

### Inv

This additional connection produces an inverted version of the envelope signal.

**The Out and Inv outputs can be used simultaneously.**

## AD (& Mod) Vintage

This envelope generator provides only two segments - Attack and Decay. Once it has started, it runs through both segments until it is finished according to the times set under Attack and Decay. It features adjustable slope for both segments, and is suitable for all kinds of modulation. There are two versions of this envelope generator. The one with the ' & Mod' designation in its name includes time modulation inputs and controls. The other does not.

### Controls

#### ASlope

Adjusts the slope curve for the attack phase. The curve is continuously adjustable from linear to logarithmic .

#### DSlope

Adjusts the slope curve for the decay phase. The curve is continuously adjustable from a linear to a exponential fade out.

#### A

The attack time. When a gate signal is received, the attack segment starts, and continues for the slope time at which point the maximum level is reached.



#### D

The decay time. The envelope enters the decay segment when the attack phase completes, and lasts for the decay slope time at which point the level reaches 0.

#### Tmod1

Adjusts the times of the two segments of the envelope. Both the intensity and direction of the modulation effect of the signal at Tmod1 is set by this value. Negative values shorten times, and positive values lengthen them. This parameter is missing on the Vintage AD & Mod version.

#### Tmod2

Adjusts the times of the two envelope segments. Both the intensity and direction of the modulation effect of the signal at Tmod2 is set by this value. Negative values shorten times, and positive values lengthen them. This parameter is missing on the Vintage AD & Mod version.

#### Lmod

Adjusts the levels of all segments of the envelope. The value here (0..max) controls the intensity of the modulation of the levels by the modulation signal.



### **Amod (Only on the AD & Mod version)**

Adjusts the time of the attack segment of the envelope. Both the intensity and direction of the modulation effect of the signal is set by this value. Negative values shorten times, and positive values lengthen them.

### **Dmod (Only on the AD & Mod version)**

Adjusts the time of the decay segment. Both the intensity and direction of the modulation effect of the signal is set by this value. Negative values shorten times, and positive values lengthen them.

## **Connections**

### **Gate**

Connect the gate signal from, for example, an MVC or a sequencer here.

### **Esync**

Connection for the Esync feedback to an MVC or sequencer module.

### **Tmod1 (Not on the AD & Mod version)**

Input for a modulation signal to control envelope times.

### **Tmod2 (Not on the AD & Mod version)**

Input for a modulation signal to control envelope times.

### **Lmod**

Input for a modulation signal to control envelope levels.

### **Amod (Not on the AD version)**

Input for a modulation signal to control attack time.

### **Dmod (Not on the AD version)**

Input for a modulation signal to control decay time.

### **Out**

The envelope's output signal.

## AHD (& Mod) Vintage

This three stage envelope generator provides Attack, Hold, and Decay times. Once started, the envelope cycle progresses through all three stages according to the times set for attack, hold and decay. It features adjustable slope parameters for attack and decay, and is suitable for any kind of modulation. There are two versions of this envelope generator - one in which each segment time can be modulated individually, as denoted by including '& Mod' in its name.

### Controls

#### ASlope

Adjusts the slope curve for the attack phase. The curve is continuously adjustable from linear to logarithmic .

#### DSlope

Adjusts the slope curve for the decay phase. The curve is continuously adjustable from a linear to a exponential fade out.

#### A

The attack time. When a gate signal is received, the attack segment starts, and continues for the slope time at which point the maximum level is reached.



#### H

The length of time to maintain the maximum level. After the attack phase has completed, the maximum level is held for the duration set here.

#### D

The decay time. The envelope enters the decay segment when the hold phase completes, and lasts for the decay slope time at which point the level reaches 0.

#### Tmod1 (Not on the AHD & Mod version)

Adjusts the times of the three segments of the envelope. Both the intensity and direction of the modulation effect of the signal at Tmod1 is set by this value. Negative values shorten times, and positive values lengthen them. This parameter is missing on the Vintage AHD & Mod version.

#### Tmod2 (Not on the AHD & Mod version)

Adjusts the times of the three envelope segments. Both the intensity and direction of the modulation effect of the signal at Tmod2 is set by this value. Negative values shorten times, and positive values lengthen them.

#### Lmod

Adjusts the levels of all segments of the envelope. The value here (0..max) controls the intensity of the modulation of the levels by the modulation signal.



### **Amod (Only on the AHD & Mod version)**

Adjusts the time of the attack segment of the envelope. Both the intensity and direction of the modulation effect of the signal is set by this value. Negative values shorten times, and positive values lengthen them.

### **Dmod (Only on the AHD & Mod version)**

Adjusts the time of the decay segment. Both the intensity and direction of the modulation effect of the signal is set by this value. Negative values shorten times, and positive values lengthen them.

### **Hmod (Only on the AHD & Mod version)**

Adjusts the time of the attack segment of the envelope. Both the intensity and direction of the modulation effect of the signal is set by this value. Negative values shorten times, and positive values lengthen them.

## **Connections**

### **Gate**

Connect the gate signal from, for example, an MVC or a sequencer here.

### **Esync**

Connection for the Esync feedback to an MVC or sequencer module.

### **Tmod1 (Not on the AHD & Mod version)**

Input for a modulation signal to control envelope times.

### **Tmod2 (Not on the AHD & Mod version)**

Input for a modulation signal to control envelope times.

### **Lmod**

Input for a modulation signal to control envelope levels.

### **Amod (Not on the AHD version)**

Input for a modulation signal to control attack time.

### **Hmod (Not on AHD version)**

Input for a modulation signal to control hold time.

### **Dmod (Not on the AHD version)**

Input for a modulation signal to control decay time.

### **Out**

The envelope's output signal.



## ADSR (& Mod) Vintage

A classic four-stage envelope generator with Attack, Decay, Sustain, and Release segments. It features separate slope parameters for Attack, and Decay/Release, and is suitable for any modulation purpose. This envelope generator is available in two versions, one of which features independent individual time and sustain level modulation (designated by including ‘& Mod’ in its name).

### Controls

#### ASlope

Adjusts the slope curve for the attack phase. The curve is continuously adjustable from linear to logarithmic.

#### DSlope

Adjusts the slope curve for the decay/release phases. The curve is continuously adjustable from a linear to a exponential fade out.

#### A

The attack time. When a gate signal is received, the attack segment starts, and continues for the slope time at which point the maximum level is reached.

#### D

The decay time. The envelope enters the decay segment when the attack phase



completes, and the modulation falls back to the sustain level. The time needed for this is the decay time. A decay will not be heard if the sustain level is set to maximum.

#### S

The sustain level. This is the level at which the modulation signal will be held as long as the gate is open. When the gate closes, the release segment immediately follows.

#### R

The release time. When the envelope generator receives a gate-off signal, it

jumps immediately from its current phase to the release stage. When the change from one segment to another takes place, the release time will be adjusted to account for the volume level at that time.

#### Tmod1 (Not on the ADSR & Mod version)

Adjusts the times of all segments of the envelope. Both the intensity and direction of the modulation effect of the signal at Tmod1 is set by this value. Negative values shorten times, and positive values lengthen them.

#### Tmod2 (Not on the ADSR & Mod version)

Adjusts the times of the three envelope segments. Both the intensity and direction of the modulation effect of the signal at Tmod2 is set by this value. Negative values shorten times, and positive values lengthen them.

#### Lmod

Adjusts the levels of all segments of the envelope. The value here (0..max) controls the intensity of the modulation of the levels by the modulation signal.





#### **Amod** (Only on the ADSR & Mod version)

Adjusts the time of the attack segment of the envelope. Both the intensity and direction of the modulation effect of the signal is set by this value. Negative values shorten times, and positive values lengthen them.

#### **Dmod** (Only on the ADSR & Mod version)

Adjusts the time of the decay segment. Both the intensity and direction of the modulation effect of the signal is set by this value. Negative values shorten times, and positive values lengthen them.

#### **Smod** (Only on the ADSR & Mod version)

Adjusts the intensity of influence of the modulation signal over the sustain level. Depending on this value, the sustain level will be modulated from 0 to the amount adjusted.

#### **Rmod** (Only on the ADSR & Mod version)

Adjusts the time of the release segment. Both the intensity and direction of the modulation effect of the signal is set by this value. Negative values shorten times, and positive values lengthen them.

## **Connections**

### **Gate**

Connect the gate signal from, for example, an MVC or a sequencer here.

### **Esync**

Connection for the Esync feedback to an MVC or sequencer module.

#### **Tmod1** (Not on the ADSR & Mod version)

Input for a modulation signal to control envelope times.

#### **Tmod2** (Not on the ADSR & Mod version)

Input for a modulation signal to control envelope times.

### **Lmod**

Input for a modulation signal to control envelope levels.

#### **Amod** (Not on the ADSR version)

Input for a modulation signal to control attack time.

#### **Dmod** (Not on the ADSR version)

Input for a modulation signal to control decay time.

#### **Smod** (Not on the ADSR version)

Input for a modulation signal to control the sustain level.

#### **Rmod** (Not on the ADSR version)

Input for a modulation signal to control release time.

### **Out**

The envelope's output signal.

## Multisegment Envelopes

The Multisegment envelope generators offer the greatest degree of freedom when working with envelopes. Two versions are provided - one with Esync, and one without (which should only be used to control pitch or filters). Otherwise the two are identical.

Up to 128 fully editable segments are available. Segments are separated by colored nodes, whereby each node marks the end of the segment behind it. The colors represent different functions. The function of a segment is not pre-defined - any segment can be defined as a Time, Time/Note Off, or Sustain segment.

Time segments always continue for the prescribed time duration, regardless of whether a Note Off message is received. These are terminated by a blue node. Time segments are used, for example, in the release phase, or to build 'one-shot' envelopes.

Time/Note Off segments terminate and the envelope proceeds to the next segment (if available) when a MIDI Note Off message is received. These segments are terminated by a red node. As you would use a Time segment for the release phase of a traditional ADSR envelope, use Time/Note Off segments in attack and decay phases.

The third type of segment is the Sustain segment. When a sustain segment is reached, the envelope remains at the sustain level until a Note Off message is received, or until a gate signal closes. When either signal is received, the envelope proceeds to the next segment, if present.

Two arbitrary points of a segment can be used to define a loop, as marked by two arrows. The number of loop iterations can be set from 1 to 256, or to Infinite. The loop will repeat while the gate is open and/or until the number of loop iterations has been reached.



The range of possibilities these envelope generators offer may appear intimidating at first. However, when you become familiar with their usage, you will appreciate the freedom they give you.

The times and levels of each envelope can be modulated, and the version with Esync also includes a variable Slope parameter.

## Controls

### Inserting and Deleting Segments

Insert a new segment by double clicking at the desired location in the envelope graphic before which you want to insert a new segment. You can double click anywhere in the envelope - at the beginning, between two segments, or at the end. A maximum of 128 segments are available. Delete a segment by double clicking on the node that identifies it.

### Selecting a Node and Changing the Mode

To select a segment node simply click on it. The node and the segment line will show as 'selected'. You can now change the mode of this segment by using the Mode button. Repeated clicking on the button walks through the modes. Continue clicking until the desired mode color shows.

## Adjusting Times and Levels

Change the time or level value of a segment by clicking on its node and dragging it. Dragging along the y axis adjusts the time, and dragging along the x axis adjusts the level. Two text fields to the left of the envelope graphic maintain a display of the current time and level of the selected envelope. You can also enter time and level values directly into the text fields.

### Defining a Loop

In order to define a loop, at least two envelope segments must be present. Enable Loop Set mode by clicking on the Set button. Click on the location where the loop is to start and a small arrow pointing right will appear to mark the beginning. Now click on the point where the loop is to end. Another small arrow, this one pointing left, will mark the end. Your loop is now defined. Delete a loop by deleting one of its markers (as above) or by using the Delete button.

## Loops

Set the number of loop repeats here. The range is from 1 to 256 repeats, or the value can be set to Infinite. A loop will repeat as long as a gate signal is open, and the number of repeats has not been exceeded. When the gate signal closes, the loop is exited and the envelope immediately proceeds to the next segment.

**Tip:** By using a segment of type Time, you can avoid having a MIDI Note Off message force an exit from the loop.

## Slope

(Not on the Modulation Multisegment Env)

You can adjust the curves for the attack, decay and release slopes. The curve is continuously adjustable from linear to logarithmic.

## Tmod1

Adjusts the modulation depth of the times of all segments of the envelope. Both the intensity and direction of the modulation signal at Tmod1 is set by this value. Negative values shorten times, and positive values lengthen them.

## Tmod2

Adjusts the modulation depth of the times of all segments of the envelope. Both the intensity and direction of the modulation signal at Tmod2 is set by this value. Negative values shorten times, and positive values lengthen them.

## Lmod

Adjusts the levels of all segments of the envelope. The value here (0..max) controls the intensity of the modulation of the levels by the modulation signal.

## Connections

### Gate

Connect the gate signal from, for example, an MVC or a sequencer here.

### Esync

(Not on the Modulation Multisegment Env)

Connection for the Esync feedback to an MVC or sequencer module.

### Tmod1

Input for a modulation signal to control envelope times.

### Tmod2

Input for a modulation signal to control envelope times.

### Lmod

Input for a modulation signal to control envelope levels.

### Out

Output for the envelope signal.

## Envelope Follower

An envelope follower tracks a dynamically changing audio signal and derives an envelope signal from it. The response time for rising levels is adjusted under Attack, and the time for falling signals under Decay. For increased control, both input and output gain are individually adjustable. A Hold function terminates analysis and freezes the signal at its current level.

### Controls

#### Input Gain

Some signal levels are too low or too high to be processed effectively. Adjust the Input Gain value up or down to optimize the signal level.

#### Attack

Controls the response time of the envelope follower to rising levels of the audio signal.

#### Decay

Controls the response time of the envelope follower to falling levels of the audio signal.



#### Output Gain

Controls the output level of the generated envelope signal.

#### Hold

Click on this button to terminate analysis and hold the envelope output at its current level.

### Connections

#### In

Input for the audio signal to be processed.

#### Out

Output for the generated envelope signal.

# Mix & Gain

This section describes the various modules used for signal mixing, switching, amplification and attenuation. Also related, and described in this group, are the VCAs for use with envelope generators and the PolyOut modules for use in polyphonic patches.

## PolyOut 1 & 2

The PolyOut modules are extremely important, as they are required in any polyphonic patch. You can think of the PolyOut as a kind of 'Master Volume' for the patch. Since a polyphonic patch consists of multiple voices, something akin to a mixer is required to combine them. This is exactly what the PolyOut modules are for.

The Gain control of these modules can actually serve as a master volume control for the patch. PolyOut 1 has a single input and a single output, and is suitable for monophonic patches. PolyOut 2 has two ins and outs, and is suitable for stereo patches. PolyOut modules are usually placed after the synthesis modules and before the effects modules in a patch.



### Controls

#### Gain

Controls the overall level of the patch. If distortion occurs, reduce the level until it is eliminated.

### Connections

#### In

Input for the audio signal.

#### Out

The audio signal output.

## Linear und Exponential VCA

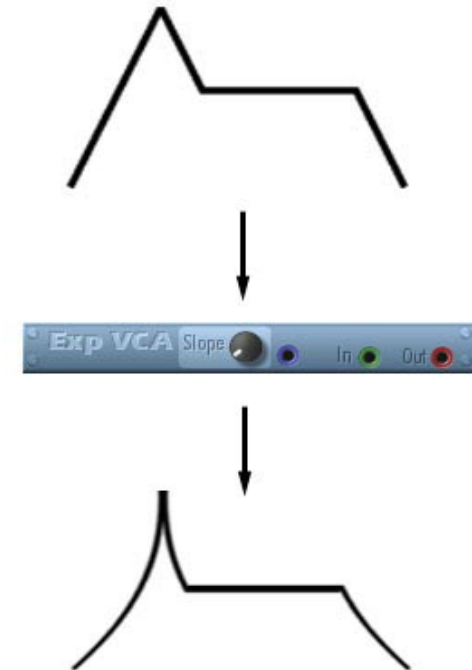
Volume control processes, such as level control using envelopes, require a VCA (**V**oltage **C**ontrolled **A**mplifier). Any audio signal can be input to the VCA. The control signal at the modulation input then controls the level of the audio signal at the output.

The Modular has the equivalent of two VCAs - linear and exponential. In the linear VCA, the level of the output is influenced by the signal at the modulation input at a ratio of 1:1. The exponential VCA takes the signal at the modulation input and processes it to produce an exponential response. A linear envelope signal becomes exponential, and the audio output level will change exponentially.

## Controls

### Slope (Exponential VCA only)

Controls how strongly the linear modulation signal is given an exponential characteristic. The curve can be adjusted continuously from linear to exponential.



## Connections

### In

Input for the audio signal.

### Mod

Input for the modulation signal.

### Out

Audio output.



## Slope Mod VCA

Basically this module corresponds to an exponential VCA, but it adds the capability to modulate the Slope setting. This is especially interesting when using arpeggio-like sequences: The slopes of the envelopes of the sounds used for the arpeggio can be varied with the help of an LFO synchronized to the tempo, thereby varying the sound character.

### Controls

#### Slope

Controls the slope curve characteristic of the incoming modulation signal. It can vary smoothly from linear to exponential.

#### Smod (Slope Modulation)

Controls the strenght of the modulation signal over the slope.



### Connections

#### In

Input for the audio signal.

#### Mod

Input for the VCA modulation signal.

#### Smod

Input for the Slope modulation signal.

#### Out

Audio signal output.

## 6dB/12dB Gain

The 6dB and 12dB Gain modules amplify a signal by up to 6dB or 12dB respectively. Amplification can lead to clipping, or distortion, which can be used as an effect if desired.



### Controls

#### Gain

Controls the amplification level of the audio signal.

### Connections

#### In

Input for the audio signal.

#### Out

Output for the amplified audio signal.

## Volume Attenuator

The Volume Attenuator can only reduce (attenuate) the signal level. To increase the level, use one of the Gain modules.



### Controls

#### Att

Adjusts the degree of signal attenuation (volume cut).

### Connections

#### In

Input for the audio signal.

#### Out

Output for the attenuated audio signal.

## Micro Mixer

This mixer has four input channels with level and pan controls per channel and a stereo output with master level control. Each input features an editable text field for labelling purposes.

**Note:** The Micro Mixer is a monophonic module, as its green color indicates. Therefore, as with monophonic effects, it should be only used following the PolyOut module.

### Controls

#### Pan 1- 4

Sets pan position for each of input signals 1-4 in the stereo output.

#### Level 1- 4

Sets the level for each of input signals 1-4 in the stereo output.



#### Master

Controls output level (left and right outputs in tandem).

#### In Name

A freely editable label field – typically, enter the name of the signal connected to each channel here.

### Connections

#### In 1- 4

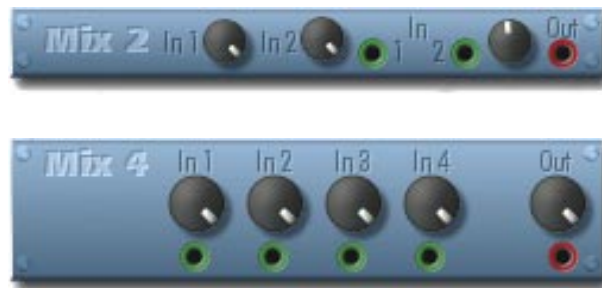
Audio inputs.

#### Master Out

Stereo (mix) output.s.

## Mix 2/4/8

These modules are simple mixers, with the number of inputs designated in their names. All inputs are mixed to a single output. All input and output levels are adjustable.



### Controls

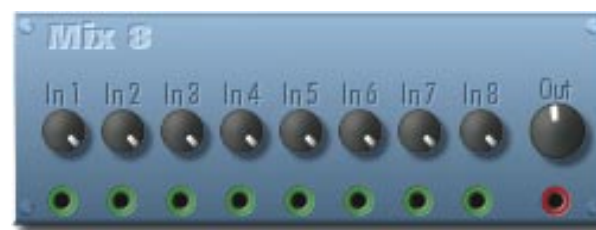
#### In

The number of input potentiometers will depend on which mixer you are using. Each one controls its respective input signal level.

**Internal distortion can occur in the mixer. If this happens, pull back on the levels a bit.**

#### Out

Controls the mixer's output level.



### Connections

#### In

Depending on the mixer, there will be 2, 4, or 8 inputs for audio signals.

#### Out

The mixer's audio output.

## Static Crossfade

Mixes two input signals to a single output. The mix ratio of the two signals is adjustable.



### Controls

#### X-Fade 1/2

Controls the relative levels of signal 1 and signal 2.

### Connections

#### In 1

Input for audio signal 1.

#### In 2

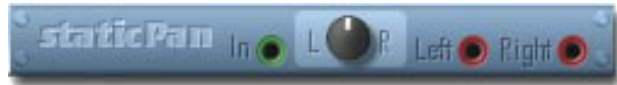
Input for audio signal 2.

#### Out

The audio output.

## Static Pan

Distributes a single audio signal to two outputs (L and R). The L/R balance is adjustable.



### Controls

#### Pan L/R

Adjusts the apportioning of the input signal to the left and right outputs.

### Connections

#### In

Input for the audio signal to be panned.

#### Out L

The left audio output.

#### Out R

The right audio output.

## Crossfade Modulator

This module mixes two input signals to a single audio output. The signal ratio can be adjusted manually, or by using a modulation source.



### Controls

#### X-Fade 1/2 (Crossfade)

Controls the relative levels of signal 1 and signal 2.

#### Mod

Controls the crossfade modulation depth.

### Connections

#### In 1

Input for audio signal 1.

#### In 2

Input for audio signal 2.

#### Mod

Input for the modulation signal.

#### Out

The audio output.

## Pan Modulator

Distributes a single audio signal to two outputs (L and R). The L/R balance is adjustable either manually, or by using a modulation signal.

### Controls

#### Pan L/R (Crossfade)

Adjusts the apportioning of the input signal to the left and right outputs.

#### Mod

Controls the depth of Pan modulation



### Connections

#### In

Input for the audio signal to be panned.

#### Mod

Input for the modulation signal.

#### Out L

The left audio output.

#### Out R

The right audio output.

# Amplitude Modulator

Use the Amplitude Modulator on any audio signal. Slow modulation results in a tremolo effect, while very fast modulation can result in spectral effects. An offset is available to add a fixed amplitude component to the modulation.

## Controls

### Mod1

Controls the modulation depth of the signal at the first modulation input.

### Mod2

Controls the modulation depth of the signal at the second modulation input.

### Offset

Adds a constant value to the amplitude modulation (affecting the resulting modulation level).

If there is no signal at the Mod1/2 inputs, you must raise the offset in order to hear the (unmodulated) signal.



## Connections

### In

Input for the audio signal.

### Mod1

Input for the first modulation signal.

### Mod2

Input for the second modulation signal.

### Out

Audio output.



### **CFadeM (Crossfade Modulation )**

Controls the intensity of the modulation of the crossfade parameter.

Modulating the crossfade time results in an effect resembling a change of the grain size in granular synthesis.

## **Connections**

### **Freq**

Input for a frequency signal. *External* must be switched on to use this signal.

### **Input 1-4**

Inputs for the 4 audio signals.

### **RMod**

Input for a modulation signal to control the LFO rate.

### **Keyf**

Input for the Note signal of the MVC.

### **CFm**

Input for a modulation signal to control crossfade time.

### **Out**

Audio output of the resulting signal.

## On/Off Switch

This provides a button in the patch with which you can interrupt (switch on and off) a signal.



### Controls

#### On/Off

This switch either allows the signal to pass through to the output or not, depending on its state. When the switch is lit, it is 'on', and the signal is routed to the output. When it is not lit, it is 'off', and the signal is not passed to the output.

### Connections

#### In

Input for the audio signal.

#### Out

Audio output.

## 1x4 Switch (Gain)

This switch accepts an audio signal and routes it through to one of four outputs. It is also available in a version that includes a gain control for each of the outputs.

### Controls

#### Switch 1 - 4

The signal will be routed to the output selected.

#### Gain 1 - 4 (1X4 Switch Gain only)

Controls the levels of the respective outputs.



### Connections

#### In

Input for the audio signal

#### Out 1- 4

Possible outputs to which the signal can be routed.

## 4x1 Switch (Gain)

This switch selects one of four audio inputs and routes it through to the output. It is also available in a version that includes a gain control for each of the inputs.

### Controls

#### Switch 1 - 4

Selects the input signal to be routed to the output.

#### Gain 1 - 4 (4X1 Switch Gain only)

Controls the levels of the respective inputs.



### Connections

#### In 1- 4

Inputs for up to four audio signals.

#### Out

Audio output.

## Peak Meter

This module allows you to easily measure an audio signal. It contains a signal level LED, a peak LED, and a text display, with reset, of the last peak level.

### Controls

#### Signal-LED

This LED lights up when the audio signal reaches or exceeds -60dB.

#### Peak-LED

This LED lights up when the signal reaches or exceeds 0dB.

#### Peak Textfield

Displays the last peak level reached.



#### Reset

The reset button clears the peak level text display so you can continue to monitor the peak levels. You would use this, for example, when the last peak reads 0dB, and you make adjustments to reduce the signal to a safe level. Click on Reset to resume monitoring the peaks.

### Connections

#### In

Input for an audio signal to measure.

# Filter

The Modular includes a number of filter modules to manipulate the frequency characteristics of audio signals. Most filters have, in addition to others, two important parameters: cutoff and resonance. Since they are so important, we'll discuss them first, along with some other important terms.

The filter cutoff frequency determines the frequencies which the filter will process, and which it will leave alone. In other words, the cutoff frequency is the point at which the filter begins to have an effect. Depending on the type of filter, the unaffected range can lie above or below (or both) the cutoff point. The range of unaffected frequencies determines whether the filter is referred to as a high-pass, band-pass, or low-pass filter. If you modulate the cutoff with, for example, an LFO or an envelope generator, the tone quality changes in response.

Also in connection with filters you will encounter the term 'Slope', expressed in dB per octave. The slope of a filter determines how strongly it will act on the frequencies beyond the cutoff. A filter slope of -12dB/Octave means that frequencies at each octave above (and/or below) the cutoff are attenuated by an additional 12dB. The higher the dB value, the steeper the slope, and the greater the attenuation.

After cutoff, resonance is the next most important parameter to consider. By feeding back a portion of the filtered signal, frequencies around the cutoff are reinforced. At high resonance values, the filter will begin to oscillate at the cutoff frequency, as that is the frequency most strongly reinforced. This is known as periodic resonance. This effect has come to be known as 'chirping', or, at really high levels, 'screeching'.

Along with the classic low/band/high-pass filter types, the Modular includes some additional filters, among them the comb filter and the vocal filter (to be described later in their own sections).

As with other modules, filters are supplied in different versions. Some have resonance, others only cutoff. Some can be controlled by modulation signals, others only by manual controls. As mentioned before, using more basic modules means a savings in DSP resources. Use the smallest version of a filter that fulfills your needs in a patch. You will be rewarded with more available voices.

## 6dB Highpass

This is a very small and economical highpass filter without resonance or modulation capabilities.



### Controls

#### Cutoff

Adjusts the cutoff frequency.

### Connections

#### In

Input for the audio signal.

#### Out

Audio output of the filter.

## 6dB Lowpass

This is a very small and economical lowpass filter without resonance or modulation capabilities.



### Controls

#### Cutoff

Adjusts the cutoff frequency.

### Connections

#### In

Input for the audio signal.

#### Out

Audio output of the filter.

## 12dB Multimode Filter

This filter has a 12dB slope, cutoff and resonance, and three parallel outputs, one for each of the basic filter types - high, band, and low pass. There is no modulation input.



### Controls

#### Cutoff

Adjusts the cutoff frequency.

#### Resonance

Controls the strength of the resonance.

### Connections

#### In

Input for the audio signal.

#### Out

Audio output of the filter.

## 18dB Lowpass

This lowpass filter has a slope of -18dB/octave, and modulation inputs. Resonance is not implemented in this filter.

### Controls

#### Cutoff

Adjusts the cutoff relative to a range of 0..127.

#### CFm1

Controls the depth of the first cutoff frequency modulation signal.

#### CFm2

Controls the depth of the second cutoff frequency modulation signal.

#### Keyf

This parameter allows the filter cutoff to track the keyboard through the MVC. The keyfollow mid-point is fixed at MIDI note #64 (E3). At this note, the cutoff frequency will always stand at its original value, regardless of the key follow



setting. When keyfollow is set to 100%, the cutoff frequency will adjust to maintain its frequency relationship to the pitch across the entire keyboard. At a setting of 50%, the cutoff frequency ratio will be lowered by 50% per octave above E3, and raised 50% per octave below E3. A value of 0% means there is no keyfollow modulation, and the cutoff frequency remains fixed.

### Connections

#### In

Input for the audio signal.

#### Keyf

Input to receive the Note signal from the MVC.

#### CFm1

Input to receive a cutoff frequency modulation signal.

#### CFm2

Input to receive a cutoff frequency modulation signal.

#### Out

The filter's audio output.



## 24dB Lowpass Filter

This low-pass filter with 24dB/octave slope behaves differently with respect to resonance. As the resonance setting is increased, the level of the original signal drops, ultimately to the point of vanishing altogether. The filter also has multiple cutoff-frequency modulation inputs.

### Controls

#### Cutoff

Adjusts the cutoff frequency relative to a range of 0..127.

#### Resonance

Adjusts the resonance relative to a range of 0..127.

#### Keyf

This parameter allows the filter cutoff to track the keyboard through the MVC. The keyfollow mid-point is fixed at MIDI note #64 (E3). At this note, the cutoff frequency will always stand at its original value, regardless of the key follow setting. When keyfollow is set to 100%, the cutoff frequency will adjust to maintain its frequency relationship to the pitch across the entire keyboard. At a setting of 50%, the cutoff frequency ratio will be lowered by 50% per octave above



E3, and raised 50% per octave below E3. A value of 0% means there is no keyfollow modulation, and the cutoff frequency remains fixed.

#### CFm1

Controls the depth of the first cutoff frequency modulation signal.

#### CFm2

Controls the depth of the second cutoff frequency modulation signal.

### Connections

#### In

Input for the audio signal.

#### ResM

Input for the modulation signal to control resonance.

#### Keyf

Input to receive the Note signal from the MVC.

#### CFm1

Input to receive a cutoff frequency modulation signal.

#### CFm2

Input to receive a cutoff frequency modulation signal.

#### Out

The filter's audio output.

## 24dB Highpass Filter

This high-pass filter with 24dB/octave slope behaves differently with respect to resonance. As the resonance setting is increased, the level of the original signal drops, ultimately to the point of vanishing altogether. The filter also has multiple cutoff-frequency modulation inputs.

### Controls

#### Cutoff

Adjusts the cutoff frequency relative to a range of 0..127.

#### Resonance

Adjusts the resonance relative to a range of 0..127.

#### Keyf

This parameter allows the filter cutoff to track the keyboard through the MVC. The keyfollow mid-point is fixed at MIDI note #64 (E3). At this note, the cutoff frequency will always stand at its original value, regardless of the key follow setting. When keyfollow is set to 100%, the cutoff frequency will adjust to maintain its frequency relationship to the pitch across the entire keyboard. At a setting of 50%, the cutoff frequency ratio



will be lowered by 50% per octave above E3, and raised 50% per octave below E3. A value of 0% means there is no keyfollow modulation, and the cutoff frequency remains fixed.

#### CFm1

Controls the depth of the first cutoff frequency modulation signal.

#### CFm2

Controls the depth of the second cutoff frequency modulation signal.

### Connections

#### In

Input for the audio signal.

#### ResM

Input for the modulation signal to control resonance.

#### Keyf

Input to receive the Note signal from the MVC.

#### CFm1

Input to receive a cutoff frequency modulation signal.

#### CFm2

Input to receive a cutoff frequency modulation signal.

#### Out

The filter's audio output.

## 24dB Lowpass Filter V

The "V" in the name stands for "vintage". This is a "classical" 24dB/octave low-pass filter in which the level of the original signal remains largely unchanged, even at high resonance settings. The filter has cutoff, resonance and various modulation inputs.

### Controls

#### Cutoff

Adjusts the cutoff frequency relative to a range of 0..127.

#### Resonance

Adjusts the resonance relative to a range of 0..127.

#### ResM

Controls the depth of resonance modulation.

#### Keyf

This parameter allows the filter cutoff to track the keyboard through the MVC. The keyfollow mid-point is fixed at MIDI note #64 (E3). At this note, the cutoff frequency will always stand at its original value, regardless of the key follow setting. When keyfollow is set to 100%, the cutoff frequency will adjust to



maintain its frequency relationship to the pitch across the entire keyboard. At a setting of 50%, the cutoff frequency ratio will be lowered by 50% per octave above E3, and raised 50% per octave below E3. A value of 0% means there is no keyfollow modulation, and the cutoff frequency remains fixed.

#### CFm1

Controls the depth of the first cutoff frequency modulation signal.

#### CFm2

Controls the depth of the second cutoff frequency modulation signal.

### Connections

#### In

Input for the audio signal.

#### ResM

Input for the modulation signal to control resonance.

#### Keyf

Input to receive the Note signal from the MVC.

#### CFm1

Input to receive a cutoff frequency modulation signal.

#### CFm2

Input to receive a cutoff frequency modulation signal.

#### Out

The filter's audio output.

## 24dB Lowpass Filter R

This low-pass filter emulates the characteristics of the filters typically found in large (and old) modular systems – hence the designation "R" (for "retro"). Its special property is that it can go into self-oscillation at high resonance settings, even in the absence of an input signal. Several modulation inputs are provided.

### Controls

#### Cutoff

Adjusts the cutoff frequency relative to a range of 0..127.

#### Resonance

Adjusts the resonance relative to a range of 0..127.

#### ResM

Controls the depth of resonance modulation.

#### Keyf

This parameter allows the filter cutoff to track the keyboard through the MVC. The keyfollow mid-point is fixed at MIDI note #64 (E3). At this note, the cutoff frequency will always stand at its original value, regardless of the key follow setting. When keyfollow is set to 100%, the cutoff frequency will adjust to



maintain its frequency relationship to the pitch across the entire keyboard. At a setting of 50%, the cutoff frequency ratio will be lowered by 50% per octave above E3, and raised 50% per octave below E3. A value of 0% means there is no keyfollow modulation, and the cutoff frequency remains fixed.

#### CFm1

Controls the depth of the first cutoff frequency modulation signal.

#### CFm2

Controls the depth of the second cutoff frequency modulation signal.

### Connections

#### In

Input for the audio signal.

#### ResM

Input for the modulation signal to control resonance.

#### Keyf

Input to receive the Note signal from the MVC.

#### CFm1

Input to receive a cutoff frequency modulation signal.

#### CFm2

Input to receive a cutoff frequency modulation signal.

#### Out

The filter's audio output.

## 24dB Highpass Filter R

This high-pass filter emulates the characteristics of the filters typically found in large (and old) modular systems – hence the designation "R" (for "retro"). Its special property is that it can go into self-oscillation at high resonance settings, even in the absence of an input signal. Several modulation inputs are provided.

### Controls

#### Cutoff

Adjusts the cutoff frequency relative to a range of 0..127.

#### Resonance

Adjusts the resonance relative to a range of 0..127.

#### ResM

Controls the depth of resonance modulation.

#### Keyf

This parameter allows the filter cutoff to track the keyboard through the MVC. The keyfollow mid-point is fixed at MIDI note #64 (E3). At this note, the cutoff frequency will always stand at its original value, regardless of the key follow setting. When keyfollow is set to 100%, the cutoff frequency will adjust to



maintain its frequency relationship to the pitch across the entire keyboard. At a setting of 50%, the cutoff frequency ratio will be lowered by 50% per octave above E3, and raised 50% per octave below E3. A value of 0% means there is no keyfollow modulation, and the cutoff frequency remains fixed.

#### CFm1

Controls the depth of the first cutoff frequency modulation signal.

#### CFm2

Controls the depth of the second cutoff frequency modulation signal.

### Connections

#### In

Input for the audio signal.

#### ResM

Input for the modulation signal to control resonance.

#### Keyf

Input to receive the Note signal from the MVC.

#### CFm1

Input to receive a cutoff frequency modulation signal.

#### CFm2

Input to receive a cutoff frequency modulation signal.

#### Out

The filter's audio output.

## 24dB Bandpass Filter R

This band-pass filter emulates the characteristics of the filters typically found in large (and old) modular systems – hence the designation "R" (for "retro"). Its special property is that it can go into self-oscillation at high resonance settings, even in the absence of an input signal. Several modulation inputs are provided.

### Controls

#### Cutoff

Adjusts the cutoff frequency relative to a range of 0..127.

#### Resonance

Adjusts the resonance relative to a range of 0..127.

#### ResM

Controls the depth of resonance modulation.

#### Keyf

This parameter allows the filter cutoff to track the keyboard through the MVC. The keyfollow mid-point is fixed at MIDI note #64 (E3). At this note, the cutoff frequency will always stand at its original value, regardless of the key follow setting. When keyfollow is set to 100%, the cutoff frequency will adjust to



maintain its frequency relationship to the pitch across the entire keyboard. At a setting of 50%, the cutoff frequency ratio will be lowered by 50% per octave above E3, and raised 50% per octave below E3. A value of 0% means there is no keyfollow modulation, and the cutoff frequency remains fixed.

#### CFm1

Controls the depth of the first cutoff frequency modulation signal.

#### CFm2

Controls the depth of the second cutoff frequency modulation signal.

### Connections

#### In

Input for the audio signal.

#### ResM

Input for the modulation signal to control resonance.

#### Keyf

Input to receive the Note signal from the MVC.

#### CFm1

Input to receive a cutoff frequency modulation signal.

#### CFm2

Input to receive a cutoff frequency modulation signal.

#### Out

The filter's audio output.



## Multimode Filter A

This filter can perform with a slope of either 12 or 24 dB/octave. It is switchable to operate as a highpass, bandpass, or lowpass filter. This filter includes cutoff and resonance parameters, as well as several modulation inputs.

### Controls

#### Cutoff

Adjusts the cutoff frequency relative to a range of 0..127.

#### Resonance

Adjusts the resonance relative to a range of 0..127.

#### ResM

Controls the depth of resonance modulation.

#### Keyf

This parameter allows the filter cutoff to track the keyboard through the MVC. The keyfollow mid-point is fixed at MIDI note #64 (E3). At this note, the cutoff frequency will always stand at its original value, regardless of the key follow setting. When keyfollow is set to 100%, the cutoff frequency will adjust to



maintain its frequency relationship to the pitch across the entire keyboard. At a setting of 50%, the cutoff frequency ratio will be lowered by 50% per octave above E3, and raised 50% per octave below E3. A value of 0% means there is no keyfollow modulation, and the cutoff frequency remains fixed.

#### CFm1

Controls the depth of the first cutoff frequency modulation signal.

#### CFm2

Controls the depth of the second cutoff frequency modulation signal.

#### Mode

Switches among highpass, bandpass, and lowpass modes. Both the knob and text field respond to user input.

#### db/Oct

Alternates between a filter slope of 12dB and 24dB. When the button is lit, the slope is set to -24dB/Oct.



## Connections

### In

Input for the audio signal.

### ResM

Input for the modulation signal to control resonance.

### Keyf

Input to receive the Note signal from the MVC.

### CFm1

Input to receive a cutoff frequency modulation signal.

### CFm2

Input to receive a cutoff frequency modulation signal.

### Out

The filter's audio output.

## Multimode Filter B

Multimode Filter B has a slope of 12dB/octave. It has three parallel outputs - one for each of the three basic filter types - high, band, and low pass. This filter includes cutoff and resonance parameters, as well as modulation inputs.

### Controls

#### Cutoff

Adjusts the cutoff frequency relative to a range of 0..127.

#### Resonance

Adjusts the resonance relative to a range of 0..127.

#### CFm1

Controls the depth of the first cutoff frequency modulation signal.

#### CFm2

Controls the depth of the second cutoff frequency modulation signal.

#### Keyf

This parameter allows the filter cutoff to track the keyboard through the MVC. The keyfollow mid-point is fixed at MIDI note



#64 (E3). At this note, the cutoff frequency will always stand at its original value, regardless of the key follow setting. When keyfollow is set to 100%, the cutoff frequency will adjust to maintain its frequency relationship to the pitch across the entire keyboard. At a setting of 50%, the cutoff frequency ratio will be lowered by 50% per octave above E3, and raised 50% per octave below E3. A value of 0% means there is no keyfollow modulation, and the cutoff frequency remains fixed.

### Connections

#### In

Input for the audio signal.

#### Keyf

Input to receive the Note signal from the MVC.

#### CFm1

Input to receive a cutoff frequency modulation signal.

#### CFm2

Input to receive a cutoff frequency modulation signal.

#### Out

The filter's audio output.

## Uknow Filter

This is one of the most complex and at the same time one of the most flexible filter in the Modular filter collection. It combines a 12dB/octave lowpass filter with a simple highpass filter in a single module. The lowpass filter includes cutoff and resonance, and can be controlled with a modulation signal. The highpass filter has an adjustable cutoff frequency.

### Controls

#### Cutoff

Adjusts the cutoff frequency relative to a range of 0..127.

#### Resonance

Adjusts the resonance relative to a range of 0..127.

#### ResM

Controls the depth of resonance modulation.

#### Keyf

This parameter allows the filter cutoff to track the keyboard through the MVC. The keyfollow mid-point is fixed at MIDI note #64 (E3). At this note, the cutoff frequency will always stand at its original value, regardless of the key follow setting. When keyfollow is set to 100%, the cutoff frequency will adjust to



maintain its frequency relationship to the pitch across the entire keyboard. At a setting of 50%, the cutoff frequency ratio will be lowered by 50% per octave above E3, and raised 50% per octave below E3. A value of 0% means there is no keyfollow modulation, and the cutoff frequency remains fixed.

#### CFm1

Controls the depth of the first cutoff frequency modulation signal.

#### CFm2

Controls the depth of the second cutoff frequency modulation signal.

#### HPF

Adjusts the cutoff frequency of the highpass filter.

### Connections

#### In

Input for the audio signal.

#### ResM

Input for the modulation signal to control resonance.

#### Keyf

Input to receive the Note signal from the MVC.

#### CFm1

Input to receive a cutoff frequency modulation signal.

#### CFm2

Input to receive a cutoff frequency modulation signal.

#### Out

The filter's audio output.

## Combfiler A/B

The comb filter gets its name from the way it filters an audio signal. A comb filter attenuates frequencies at several regular frequency intervals. If you look at a signal that has been processed by a comb filter in a spectrum analyzer, you will notice several 'notches' where the signal has been cut. A graphic of the frequency response strongly resembles a comb, hence the name.

The Modular provides two versions of the comb filter. Comb Filter A cuts large swaths into the frequency spectrum, allowing rather narrow frequency bands to emerge after filtering. Comb Filter B cuts steep notches into the spectrum, allowing more frequencies to pass through. Even so, the two filters sound quite similar.

**Important: To hear the effect of the comb filter, you must use resonance.**



### Controls

#### Cutoff

Adjusts the cutoff frequency.

#### Resonance

Adjusts the resonance level. Increasing this value increases the filter effect.

#### RmA

Controls the depth of resonance modulation.

#### Damp

The resonance of the filter is created using feedback. By adjusting the Damp parameter, you control the depth of the feedback loop. This provides an additional way to control the behavior of the comb filter.

#### DampMod

Controls the depth of the modulation signal for Damp modulation.

## **Keyf**

This parameter allows the filter cutoff to track the keyboard through the MVC. The keyfollow mid-point is fixed at MIDI note #64 (E3). At this note, the cutoff frequency will always stand at its original value, regardless of the key follow setting. When keyfollow is set to 100%, the cutoff frequency will adjust to maintain its frequency relationship to the pitch across the entire keyboard. At a setting of 50%, the cutoff frequency ratio will be lowered by 50% per octave above E3, and raised 50% per octave below E3. A value of 0% means there is no keyfollow modulation, and the cutoff frequency remains fixed.

## **CFm1**

Controls the depth of the first cutoff frequency modulation signal.

## **CFm2**

Controls the depth of the second cutoff frequency modulation signal.

## **Connections**

### **In**

Input for the audio signal.

### **Keyf**

Input to receive the Note signal from the MVC.

### **CFm1**

Input to receive a cutoff frequency modulation signal.

### **CFm2**

Input to receive a cutoff frequency modulation signal.

### **Out**

The filter's audio output.

## Vocal Filter

This module filters signals such that the formants for the vowels A, E, O - and many others - are applied. You can select from among 10 vowel sounds distributed over 5 positions. By stepping through the positions (selected vowels), which can be done via modulation, the filter begins to 'speak'. The vocal filter also offers adjustable resonance, and a frequency offset for shifting the formants.

## Controls

### Vocal

The most important settings, the five Vocal Positions, are found on the light blue field. Each of the five positions can be adjusted to one of the 10 vowel sounds using the text fader. The rotary control in the center sets which vowel is current, and/or the starting position for modulation. The vowel the white line on the control points to is produced by filtering.

The following vowel sounds are available: A, E, I, O, U, Y, AA, AE, OE, and UE.

### Resonance

Resonance adds and strengthens the formant for the vowel sound.



### VPKeyf (Vocal Position Keyfollow)

Controls the influence of the MVC note signal on the Vocal Position. The center key is fixed at MIDI note number 64 (E3). At this note there is no keyfollow modulation, and the vowel indicated by the white line on the Vocal Position rotary control will be heard. The intensity and direction of the keyfollow modulation is adjustable from -200% to +200%.

To distribute the five possible vowels over the entire MIDI note range (C-2 to G8), you must set the Vocal Position control to vowel 3, and adjust VPKeyf to +100%. When VPKeyf is adjusted to +50%, only three vowels are distributed over the MIDI range. At 0% there is no modulation, and the vowel pointed to by the Vocal Position control will apply to the entire note range.

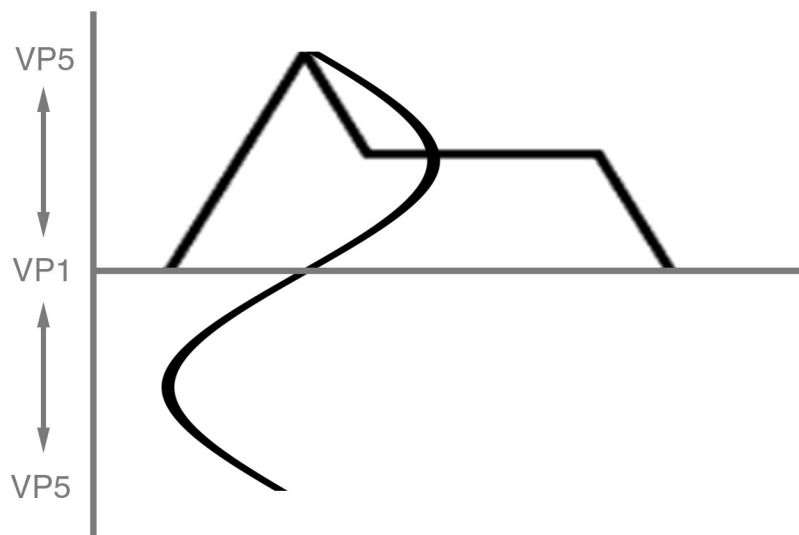
### VPm1

Adjusts the strength of the effect of the modulation signal over the Vocal Position.

### VPm2

Adjusts the strength of the effect of a second modulation signal over the Vocal Position.

**It is important to differentiate between the modulation signals coming from envelope generators, and those from LFOs, which are connected to the VPm1/2 inputs.**



**Envelopes** are unipolar modulation signals. The modulation starts at the vowel adjusted with the Vocal Position control and proceeds to step through the vowels depending on the adjusted intensity.

**LFOs** deliver bipolar modulation signals. When the wave is in the positive range, the signal is processed like an envelope - it starts at the current vowel position and proceeds through following vowels depending on the modulation intensity. When the wave enters the negative range, however, the modulation steps through the vowels in reflected, or reverse, order.

### **Freq Offset**

The Frequency Offset shifts the formant higher on the frequency scale. The result is a sonic change from a dark to a brighter sounding vowel.

### **FM**

Controls the strength of the modulation signals over the Frequency Offset.

## **Connections**

### **In**

Audio signal input.

### **VPKeyf**

Input for the Note signal from an MVC.

### **VPm1**

Input for a Vocal Position modulation signal.

### **VPm2**

Input for a second Vocal Position modulation signal.

### **FM**

Input for a Frequency Offset modulation signal.

### **Out**

Output for the filtered signal.

## 12dB Lowshelf EQ

This filter cuts or boosts frequencies below the cutoff frequency. The slope is equivalent to 12dB, but the curve resembles a 'cow's tail'.



### Controls

#### Freq

Adjusts the cutoff frequency.

#### Gain

Controls the degree of cut or boost of the frequencies below the cutoff.

#### Bypass

Sends the input signal directly to the output, bypassing the EQ.

### Connections

#### In

Input for the audio signal.

#### Out

Output for the filtered signal.

Modular

## Parametric EQ

With the Parametric EQ you can boost or cut frequencies in a range surrounding the cutoff frequency. The slope is the equivalent of 12dB, and the curve is bell-shaped. A Q-factor controls the 'quality' of the filter - i.e. the frequency range over which the filter works.

### Controls

#### Freq

Adjusts the cutoff frequency.

#### Gain

Controls the amount of cut or boost of the frequencies around the cutoff frequency.

#### Q

Increasing this value reduces the range of frequencies around the cutoff frequency to be filtered. Low values create a gentle slope, but affect more frequencies. Use a high 'Q' to 'notch out' unwanted frequencies.

#### Bypass

Sends the input signal directly to the output, bypassing the EQ.



### Connections

#### In

Input for the audio signal.

#### Out

Output for the filtered signal.

Contents

Index

72



## 12dB Highshelf EQ

This filter cuts or boosts frequencies above the cutoff frequency. The slope is equivalent to 12dB, but the curve resembles a 'cow's tail'.



### Controls

#### Freq

Adjusts the cutoff frequency.

#### Gain

Controls the degree of cut or boost of the frequencies above the cutoff.

#### Bypass

Sends the input signal directly to the output, bypassing the EQ.

### Connections

#### In

Input for the audio signal.

#### Out

Output for the filtered signal.

## Free Filter Bank A/B

These permit individual frequency bands to be boosted or cut. The frequency of each filter can be freely set, hence the name. Version A has twelve bands, version B only five. Both feature Q controls affecting all bands and output gain controls as well as a separate built-in preset list.

### Controls

#### Freq text fields

Enter frequency values for each band via these fields.

#### Freq Gains

Sets the boost or cut for each band.

#### Filter Q

Adjusts the Q (filter sharpness) of all bands in common.

#### Gain

Adjusts output level.

#### Bypass

Sends the input signal directly to the output, bypassing the filters.



### Connections

#### In

Audio signal input.

#### Out

Audio signal output.

## LFO

Along with envelope generators, LFOs (**L**ow **F**requency **O**szillatoren) are the most important modules available to control sound parameters in a patch. LFOs are special oscillators - they operate at very low frequencies. Their signals are used for periodic modulation of various parameters. The final modulation effect depends on the waveform used. Sine and triangle waves produce smooth, flowing modulation. On the other hand, rectangle or sample & hold settings produce a 'jumpy' effect.

Most modulation techniques produce characteristic effects such as tremolo (amplitude modulation), vibrato (pitch modulation) and 'wah-wah' (filter cutoff modulation). SCOPE Fusion Platform's LFOs can do all these, but there's more. Some of the LFOs use small, internal envelope generators to fade the modulation in or out - synchronously. And a key follow feature allows an LFO's frequency to track the keyboard. LFOs with an External In can be used to simulate an ordinary, audio oscillator.

A Random Signal Generator and a Sample & Hold are also included in this section.

## Single/Poly und LFOs

When using LFOs it is particularly worth considering whether to use a monophonic or polyphonic module. When a polyphonic LFO is loaded, it is loaded into the DSPs once for each voice. Modulation settings are thus implemented per voice. If the LFOs are not synchronized to the keyboard, rich modulations can develop as the various LFOs vary in phase.

If you are looking for a more traditional, simple modulation effect, load a monophonic LFO. This will load into the DSPs only once, and will serve to modulate all targets simultaneously. This also saves significant DSP overhead, leaving more room for additional voices or modules.

## Multi LFO A

Of all SCOPE Fusion Platform's LFOs, this is the most complex. Available waveforms are sine, triangle, square, saw up, saw down, and sample & hold. The frequency can be furnished either internally or externally, and can be modulated. The waveform can be retriggered, and the modulation can be controlled by a simple internal envelope generator.

### Controls

#### Rate

The frequency/rate of the modulation. The frequency, in cycles per second, is displayed in an associated text field.

#### Ext

Switches on the Freq input. The frequency is now controlled by the signal at the Freq input, and the Rate control is no longer active.

#### Waveform

Selects the desired waveform.

#### Retrig

This switch determines whether the signal will run continuously, or be restarted at its initial phase setting each time a new note is played. Retrigger is active when the button is lit.



#### Init Phase

Determines the position within the waveform (phase) at which the signal will start when a gate signal is received. Retrigger must be enabled for this to take effect.

#### Delay

Delays the onset of the modulation. A gate signal must be connected. The range is from 0 to 20 seconds.

#### Fade In

When a gate signal is received, the modulation will gradually build to maximum at the time set here. A gate signal must be connected. The range is from 0 to 20 seconds.

Between the fade in and the fade out, the modulation remains at maximum.

#### Fade Out

When a gate off signal is received, the modulation will gradually fade to 0 at the time set here. A gate signal must be connected. The range is from 0 to 20 seconds.

#### Rmod1

Controls the modulation depth of the Rate (frequency) of the LFO.

#### Rmod2

Controls the modulation depth of a second signal providing Rate (frequency).

## Keyf

This parameter allows the Rate (frequency) to track the keyboard through the MVC. The keyfollow mid-point is fixed at MIDI note #64 (E3). At this note, the cutoff frequency will always stand at its original value, regardless of the key follow setting. When keyfollow is set to 100%, the LFO frequency adjusts to maintain its relationship to the pitch across the entire keyboard as it follows the pitch. At a setting of 50%, the frequency ratio will be lowered by 50% per octave above E3, and raised 50% per octave below E3. A value of 0% means there is no keyfollow modulation, and the LFO rate remains fixed.

## Connections

### Freq

Input for a frequency signal. External mode must be enabled to use this input.

### Gate

Connect a gate signal for use with Retrigger, Delay, Fade In and Fade Out.

### Rmod1

Input for a modulation signal (Rate).

### Rmod2

Input for a second modulation signal (Rate).

### Keyf

Input for a Note signal from the MVC.

### Out

LFO signal output.

## Multi LFO B

This LFO is similar to Multi LFO A. Like Multi LFO A, available waveforms are sine, triangle, square, saw up, saw down, and sample & hold. The frequency can be furnished either internally or externally. The waveform can be retriggered, and the modulation can be controlled by a simple internal envelope generator. Because this LFO does not have Rate Modulation, it is somewhat more economical to use than Multi LFO A.

### Controls

#### Rate

The frequency/rate of the modulation. The frequency, in Hz, is displayed in an associated text field.

#### Ext

Switches on the Freq input. The frequency is now controlled by the signal at the Freq input, and the Rate control is no longer active.

#### Waveform

Selects the desired waveform.

#### Retrig

This switch determines whether the signal will run continuously, or be restarted at its initial phase setting each time a new note is played. Retrigger is enabled when the button is lit.



#### Init Phase

Determines the position within the waveform (phase) at which the signal will start when a gate signal is received. Retrigger must be enabled for this to take effect.

#### Delay

Delays the onset of the modulation. A gate signal must be connected. The range is from 0 to 20 seconds.

#### Fade In

When a gate signal is received, the modulation will gradually build to

maximum at the time set here. A gate signal must be connected. The range is from 0 to 20 seconds.

**Between the fade in and the fade out, the modulation remains at maximum.**

#### Fade Out

When a gate off signal is received, the modulation will gradually fade to 0 at the time set here. A gate signal must be connected. The range is from 0 to 20 seconds.

### Connections

#### Freq

Input for a frequency signal. External mode must be enabled to use this input.

#### Gate

Connect a gate signal for use with Retrigger, Delay, Fade In and Fade Out.

#### Out

LFO signal output.

## MW LFO

This LFO produces very simple modulations. It uses only one waveform (triangle) and includes retrigger, delay, and fade in capabilities. Since it conforms to modulation wheel standards, it is best used to provide vibrato to a synthesizer sound.

### Controls

#### Rate

The frequency/rate of the modulation. The frequency, in cycles per second, is displayed in an associated text field..

#### Retrig

This switch determines whether the signal will run continuously, or be restarted at its initial phase setting each time a new note is played. Retrigger is enabled when the button is lit.

#### Init Phase

Determines the position within the waveform (phase) at which the signal will start when a gate signal is received. Retrigger must be enabled for this to take effect.



#### Delay

Delays the onset of the modulation. A gate signal must be connected. The range is from 0 to 20 seconds.

#### Fade In

When a gate signal is received, the modulation will gradually build to maximum at the time set here. A gate signal must be connected. The range is from 0 to 20 seconds.

### Connections

#### Gate

Connect a gate signal for use with Retrigger, Delay, and Fade in.

#### Out

LFO signal output.

## Pulse LFO

This is another rather complex LFO. It produces pulse wave modulation, so it is useful for trill and doubling effects. The pulse width can be set manually, or modulated. The frequency can be furnished either internally or externally, and can also be modulated. Retriggering by a gate signal is also implemented.

### Controls

#### Rate

Sets the rate (frequency) of the LFO. The value in cycles per second is displayed in the associated text field.

#### Ext

Switches the Freq input on. The frequency is now taken from the signal connected to the Freq input. The Rate control is disabled when Ext is switched on.

#### Retrig

Makes synchronization and/or waveform restarting with each keystroke possible when enabled (button is lit).



#### Init Phase

Determines the position within the waveform (phase) at which the signal will start when a gate signal is received. Retrigger must be enabled, and a gate signal connected, for this to take effect.

#### PWidth

Control to manually adjust the pulse width.

#### PwmA

Controls the modulation depth of the signal connected to the Pwm input to modulate the pulse width.

#### Rmod1

Input for a modulation signal (Rate/frequency).

#### Rmod2

Input for a second modulation signal (Rate/frequency).



## Keyf

This parameter allows the Rate (frequency) to track the keyboard through the MVC. The keyfollow mid-point is fixed at MIDI note #64 (E3). At this note, the cutoff frequency will always stand at its original value, regardless of the key follow setting. When keyfollow is set to 100%, the LFO frequency adjusts to maintain its relationship to the pitch across the entire keyboard as it follows the pitch. At a setting of 50%, the frequency ratio will be lowered by 50% per octave above E3, and raised 50% per octave below E3. A value of 0% means there is no keyfollow modulation, and the LFO rate remains fixed.

## Connections

### Freq

Input for a frequency signal. External mode must be enabled to use this input.

### Gate

Connect a gate signal for use with retrigger.

### Rmod1

Input for a modulation signal (Rate).

### Rmod2

Input for a second modulation signal (Rate).

### Keyf

Input for a Note signal from the MVC.

### Out

LFO signal output.

## Saw Down LFO

A descending sawtooth LFO in which only the rate can be adjusted.



### Controls

#### Rate

Adjusts the rate (frequency) of the LFO.

### Connections

#### Out

LFO output.

## Saw Up LFO

An ascending sawtooth LFO in which only the rate can be adjusted.



### Controls

#### Rate

Adjusts the rate (frequency) of the LFO.

### Connections

#### Out

The LFO output.

## Sinus LFO

A sine wave LFO in which only the rate can be adjusted.



### Controls

#### Rate

Adjusts the rate (frequency) of the LFO.

### Connections

#### Out

The LFO output.

## Square LFO

A square wave LFO in which only the rate can be adjusted.



### Controls

#### Rate

Adjusts the rate (frequency) of the LFO.

### Connections

#### Out

The LFO output.

## Triangle LFO

A triangle wave LFO in which only the rate can be adjusted.



### Controls

#### Rate

Adjusts the rate (frequency) of the LFO.

### Connections

#### Out

The LFO output.

## Random Signal Generator

This signal generator produces three random signals. Sine is sinusoidal, Tri is triangular, and Step is a well-behaved 'Sample & Hold' signal. Each signal varies in amplitude and frequency. The level of amplitude variation is adjustable. The basic frequency used to produce the random values can be supplied internally or externally.

### Controls

#### Rate

Adjusts the basic frequency from which random values are derived.

#### Ext

Switches on the Freq input. When enabled, the basic frequency is controlled by the signal at Freq, and the Rate control no longer has any effect.

#### Level

Controls the amplitude of the Sine, Tri, or Step waves.



### Connections

#### Freq

Input for a frequency signal. Ext must be switched on.

#### Sine

Output for the sinusoid generated signal.

#### Tri

Output for the triangle generated signal.

#### Step

Output for the 'sample & hold' generated signal.

## Sample & Hold

This module 'probes' (samples) the signal connected to the audio input to determine the amplitude, and holds the value until the next sample. The master clock for sampling the amplitude can be an LFO, or a gate, etc. If the source signal is noise, then this becomes a random number generator which generates a random level with each gate event or LFO period.

### Controls

#### Gate/Trigger

Input for the signal to be used as a master clock.

#### Threshold

Here you can adjust a threshold level for the signal at the Audio-Trig input. When the signal exceeds the threshold, a clock pulse is generated to capture a new sample.

**Tip:** Use the threshold setting, for example, to trigger a new sample from a certain level in the attack phase of an envelope.



### Connections

#### In

Input for an audio signal from which the instantaneous value of the amplitude will be sampled, and held. You can use white noise for random effects, or for interesting patterns use a sine or triangle wave (in conjunction with an LFO master clock).

#### Gate

Connect the gate signal from an MVC or a sequencer here to use it as a master clock for sampling the input waveform.

#### Audio Trig

Connect a modulation signal here to use it as a master clock. The sample probing will be triggered depending on the nature of the signal and the threshold setting.

#### Out

Output for the Sample & Hold signal.

## Modifiers

The modules in the Modifier group are used mainly with frequency and modulation signals. For example, several pitch modifiers are available which you can insert between the MVC and an oscillator. You can then connect a modulation signal to the modifier to control the intensity of the pitch modulation.

If you want to modulate several oscillators at the same time, simply connect them all to a single modifier (see the figure beside).

If you need to provide an oscillator with a fixed frequency, or a frequency offset, use one of the 'Constant Value' modules available for the different signal classes. These modules can also be used to change patch values via MIDI-Ctrl.



## Pitch Modifier A

This is the most versatile of the pitch modifiers. It features Coarse and Fine tuning controls, and three modulation inputs for pitch modulation.

### Controls

#### Coarse/Fine

Controls the pitch of the connected frequency signal. Coarse adjusts the pitch in semitones; Fine adjusts it in cents (1/100 of a semitone).

#### PMod1

Controls the modulation depth of the signal at the first pitch modulation input. The source could be an envelope generator, LFO etc.

#### PMod2

Controls the modulation depth of the signal at the second pitch modulation input. The source could be an envelope generator, LFO etc.



#### Keyf

Controls the influence of the value of the Note signal from an MVC over the frequency of the connected oscillator(s). The MIDI note number at which there is no modulation (i.e. the center value) is fixed at #64 (E3). If key follow is adjusted to 100%, then the oscillator frequency will correspond to the played note. When adjusted to 50%, the frequency of the oscillator will be reduced by 50% per octave of the original frequency. For notes below E3, the frequency will be raised by 50% per octave. If adjusted to 0% the pitch will be the same across the entire keyboard.

### Connections

#### Freq In

Input for a frequency signal (as from an MVC).

#### PMod1

Input for the first pitch modulation signal.

#### PMod2

Input for the second pitch modulation signal.

#### Keyf

Input for the Note signal from an MVC.

#### Freq Out

Not 'freak out'. Output for the modifier's frequency signal.

## Pitch Modifier B

This modifier has only two modulation inputs, but is sufficient for most pitch modulation purposes.

### Controls

#### PMod1

Controls the modulation depth of the signal at the first pitch modulation input. The source could be an envelope generator, LFO etc.

#### PMod2

Controls the modulation depth of the signal at the second pitch modulation input. The source could be an envelope generator, LFO etc.



### Connections

#### Freq In

Input for a frequency signal (as from an MVC).

#### PMod1

Input for the first pitch modulation signal.

#### PMod2

Input for the second pitch modulation signal.

#### Freq Out

Output for the modifier's frequency signal.



## Pitch Modifier C

This modifier features two modulation inputs with special characteristics. One input is linear, and the other, exponential. The signal at the linear input adds modulation to the frequency signal. The signal at the exponential input by its nature is limited to  $\pm 36$  semitones. You must limit the input signal to this fixed range.

### Controls

#### LinM

Controls the modulation depth of the linear pitch modulation signal. Possible modulation sources include envelope generators, LFOs, etc.

#### ExpM

Controls the modulation depth of the exponential modulation signal. Possible sources include envelope generator's, LFOs, etc. The text field displays the modulation amount in semitones, and is limited to a range of  $\pm 36$  semitones.



**Example: The modulation always influences the original frequency value as a center frequency. A setting of 12, for example, implies that modulation of  $\pm 12$  semitones is in effect. This equals 24 semitones. If you now set the value to -12 (note that the sign has changed), the range is still 24 semitones, but the phase of the modulation has been adjusted by 180 degrees.**

### Connections

#### Freq In

Input for a frequency signal (e.g. from an MVC).

#### LinM

Input for a modulation signal.

#### ExpM

Input for a modulation signal.

#### Freq Out

Output for the modulated frequency signal.

## Constant Value

Use this module to provide control values for modulation inputs. It is normally used with unipolar (positive value) inputs but can also be used with bipolar modulation inputs. However, only the positive component of the modulation will be regulated, as the Constant Value module does not produce negative values. Constant Value can also be used with a MIDI controller to produce modulation values via MIDI.



### Controls

#### Val

Specify the modulation factor relative to a scale in the range of 0..127.

### Connections

#### Out

The output of the unipolar signal.

## Constant Value bipolar

Use this module to provide control values for modulation inputs. It is normally used with bipolar (positive and negative value) inputs but can also be used with unipolar modulation inputs. However, only the positive component of the modulation will be regulated, as the unipolar input treats negative values as positive values. Constant Value Bipolar can also be used with a MIDI controller to produce modulation values via MIDI.



### Controls

#### Val

Specify the modulation factor relative to a scale in the range of -64..+63.

### Connections

#### Out

The output of the bipolar signal.

## Constant Freq

This module produces a constant frequency signal which can be connected to an oscillator, or a frequency divider.



### Controls

#### Freq

Adjusts the value of the frequency signal. The text field displays the value in Herz.

### Connections

#### Out

The output of the frequency signal.

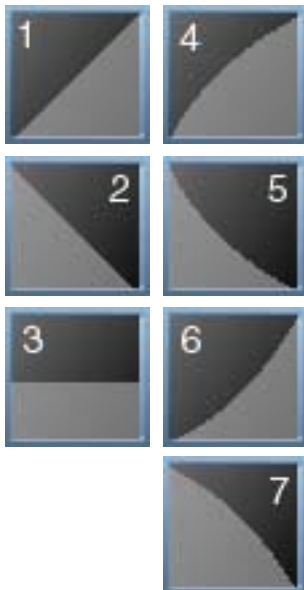
## Curve Table

The Curve Table graphic represents the 128 possible values of the MVC's Note, Vel, and At outputs. 'Massaged' values are output according to the curve in the graphic representation. Using the Curve, Sensitivity and Offset settings, you can adapt the behavior of incoming values as you desire. Seven curves determine the fundamental behavior - e.g. Linear (curve 1), Fixed (curve 3), Exponential (curve 4), or logarithmic (curve 6). The Sensitivity parameter will fit the curve to achieve the desired result, and the Offset will add or subtract a constant.

### Controls

#### Curve

Select one of the curves in the graphic to define the basic behavior.



#### Sensitivity

Adjusts the upward slope of the curve, and the resulting output values as displayed in the graphic.

#### Offset

Adds or subtracts the value set here.

### Connections

#### Val

Input for the MVC's Note, Vel, or At outputs.

#### Out

The modulation signal output.

## Linear Scale

This module processes the values of the Note, Vel and AT outputs of the MVC. Its functionality corresponds to that of a classic Keyfollow, except that in this module an offset can be entered if desired (see digram). Use this module for modulating Pan, Crossfade or Filter Cutoff using Note or Velocity signals.

### Controls

#### Scale

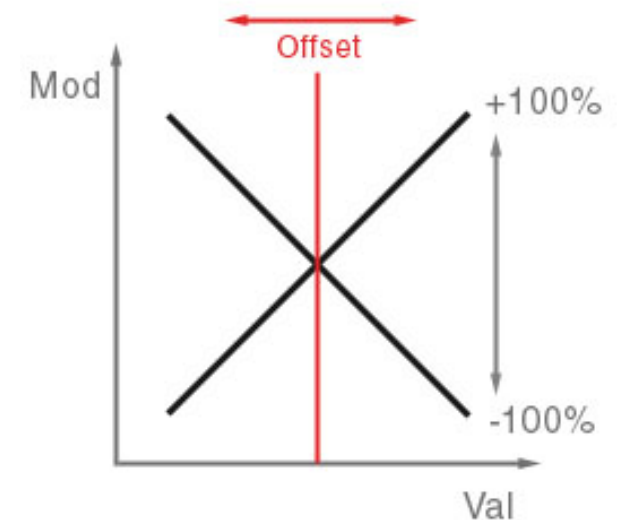
Controls the modulation response to incoming Note, Vel or AT MVC signals relative to their numeric value. The center value for scaling is MIDI value 64 (Note=E3). This value produces no scaling modulation. If the value is above or below this center value, a number will be added or subtracted to the modulation signal, depending on the Scale setting. If Scale is adjusted to 100%, an incoming signal is output at the Note, Vel and AT outputs at full modulation when the input is at its maximum value. A setting of 50% means that the degree of modulation will be scaled to produce only up to 50% of the original modulation. A value of 0% means that Scale reduces the modulation signal to nothing. Negative values reverse the direction of the modulation.



**Example:** You are modulating Pan using Note values. Linear Scale is set to +100%. If you have a keyboard with 128 keys, the lowest note will sound at the far left, and the highest note at the far right. Now, if you adjust Scale to -100%, the reverse is true.

#### Offset

Use this parameter to shift the center position for scaling by -64 to +64. The resulting modulation shifts correspondingly.



## Frequency Divider

With this module you can derive multiples or divisions of an input frequency. It provides an input for the basic frequency, and four outputs for multiples or divisions of the basic frequency. Although the module is intended to operate with the MIDI clock, it can also work alone. With a MIDI clock you could, for example, derive several different frequencies and send them in parallel to several LFOs. You could also use the Constant Freq module and similarly output several frequencies to the LFOs.

### Controls

#### Freq Divide

For each of the four outputs there is a text fader to adjust the multiplication or division factors (see the example beside).



### Connections

#### Clock

Input for the frequency signal from a MIDI clock or another module.

#### Out 1

Frequency output for the first frequency division.

#### Out 2

Frequency output for the second frequency division.

#### Out 3

Frequency output for the third frequency division.

#### Out 4

Frequency output for the fourth frequency division.

#### Example:

You have a MIDI Clock, a Frequency Divider, and two LFOs. The Freq output of the MIDI Clock supplies a value of 2.00 cycles per second at 120 BPM which we will take to be half the period of a quarter note. You want the first LFO to correspond to a quarter note, and the second to a half note triplet. Adjust Out 1 to read 24/12, and Out 2 to read 24/8. Out 1 will now produce a frequency equal to a quarter note as 24/12 equals twice the original frequency, or 4.00 cycles per second, or four oscillations per clock. Out 2 produces half note triplets, because  $24/8 = 3.00$  cycles per second, or 3 oscillations per clock. The following table contains other clock values and the corresponding se

$24/96 = 1/1$  note       $24/8 = 1/8$  triplets

$24/48 = 1/2$  note       $24/6 = 1/16$  note

$24/24 = 1/4$  note       $24/4 = 1/16$  triplets

$24/16 = 1/4$  triplets       $24/3 = 1/32$  note

$24/12 = 1/8$  note

## Frequency Multiply

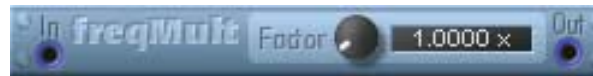
Use this module with oscillators, etc. to produce harmonics of the fundamental frequency. Harmonics are integral multiples of a fundamental frequency, and are required for additive synthesis. This module allows for non-integral decimal factors, which produce non-harmonic frequencies.

### Controls

#### Partial

This is the factor by which the basic frequency at Freq In is multiplied. By adjusting this value to integer (whole number) values, harmonics are produced: A factor of 1.000x produces the basic frequency, or first harmonic. A factor of 2.000x produces the octave above the basic frequency, or the second harmonic, etc.

Since using the rotary control to adjust this value is a little clumsy, it is better to simply enter the desired harmonic number in the text field directly. For example, enter <3> + <Enter> to adjust the oscillator to the third harmonic.



### Connections

#### Freq In

Input for a frequency signal, as from an MVC.

#### Freq Out

Output for the multiplied frequency signal.

## Pitch Quantizer

This module allows you to derive discrete pitch values from a continuous modulation signal. You can choose the desired interval and range to use for quantizing.

### Controls

#### Range

Specify the range, in semitones, within which the signal will be quantized.

#### Quant

Adjusts the quantization interval in semitones. A value of 1 equals a semitone, 2, a whole tone, 3, a minor third etc. The maximum allowable value depends on the range setting. If the range is adjusted to +/-24 semitones, then the maximum interval is 24 semitones.



### Connections

#### In

Input for the modulation signal to quantize.

#### Out

Output for the quantized signal.

## Xmod & Feedback Connector

It can sometimes happen in a patch that you will create a feedback loop (recursive connection). This could happen, for instance, when crossmodulating oscillators. However, even if the signals are compatible, SCOPE Fusion Platform does not allow for direct connection of signals if it creates a loop. If you are designing a patch in which modules are to modulate each other mutually, and you get an error message, try inserting this module between the modules. It should now work.

### Connections

#### In

Input for a modulation signal.

#### Out

Output for a modulation signal.





# Drum

The Drum oscillators permit the production of analog-style drum sounds with the Modular. Since you can load each module several times, entire drum kits are possible. Each module offers an abundance of parameters with which to create a great variety of sounds. In order that you do not lose the sounds you create with the many new parameters, we have supplied the most important modules with their own Preset lists. With the self-contained Presets, you can store your sounds immediately. With the Event Sequencer MDS8 you can create drum grooves of up to eight instruments.



## Drum Synth

The Drum Synth is the most versatile of the drum oscillators. It is designed in three parts: Sine1, Sine2, and Noise. Sine1 is the Master over Sine2. In other words, Sine1 determines the pitch of both oscillators. The Noise section features a filter to produce, in addition to analog hi-hat sounds, varieties of 'Bips' and 'Bleeps' that may come in handy at times.

This module is quite resource hungry, and you should use it only for very complex drum sounds. For straightforward sounds, the Modular offers the Drum Oscillator and the Percussion Oscillator (see later).

This module offers its own Preset list in addition to the Modular's Preset list. Use the existing presets, or store your own collection.

### Controls

#### Presets

Opens the module's Preset list.

**Note:** Presets can play back at substantially high volumes. Therefore use a moderate volume level when auditioning them. Otherwise damage to your audio gear and/or your hearing could occur.



## Note No.

Assigns the MIDI note number to which the oscillator will respond.

**Note: This setting is not saved with a preset, but only with the patch. This way when you develop a drum map with several oscillators, and you want to change sounds, the key assignments remain appropriate for the song.**

## FVel

Assigns a velocity value to use when the sound is triggered by a gate signal.

**This value is effective only when the DrumSynth is triggered by Gate. When it is triggered via MIDI, the MIDI note-on velocity is used instead.**

## Sine 1 Decay

Sets the time it takes for Sine 1 to fade to silence after being triggered.

## Sine 1 809

Adjusts smoothly from a normal decay envelope to a hold-decay envelope. Turning the control to the right increases the 'hold' part of the envelope while reducing the 'decay' portion. The envelope time remains unchanged.

## Sine 1 Snap

Adds an additional 'click' to the attack to further define the striking of a drum.

## Sine 1 Tune

Tunes the basic pitch of the instrument. Sine 1 is the master of Sine 2, so Sine 2 does not have a tuning capability, except Detune. This detunes it from Sine 1.

## Sine 1 PMod

Adjusts the pitch modulation depth.

## Sine 1 PDec

Sets the duration of the pitch modulation. Depending on the PMod and PDec settings, you can add character to the sound anywhere from 'boomy' to 'scratchy'.

## Sine 2 Decay

Sets the time it takes for Sine 2 to fade to silence after being triggered.

## Sine 2 Detune

Detunes Sine 2 relative to Sine 1. Since Sine 1 is the 'master' oscillator, the overall tuning of the instrument is governed by Sine 1 alone.

**Tip: This is especially interesting to use with tom toms. Subtle detuning can significantly enhance the realism of the sound.**

### **Sine 2 Slope**

Adjusts the decay characteristic of Sine 2. When set to the left, the decay is linear (slower fadeout). When set to the right, the fade is exponential (relatively quick). Percussive sounds naturally fade exponentially, so set the control towards the right for percussive sounds.

### **Noise Attack**

Controls the transient noise. Longer attack times allow the tone to carry through a little more. Only when the attack phase has completed will the decay phase begin.

### **Noise Decay**

Controls the time it takes the noise to fade out after the attack phase.

### **Noise Cf**

Adjusts the cutoff frequency of the noise filter, and hence the tone of the noise. At high resonance you can use the cutoff control to determine the basic pitch of, for example, a bass drum.

### **Noise Res**

Adds filter resonance. Using this control alone you can generate filtersnaps or bass drums!

### **Noise Cf Mod**

Adjusts the cutoff frequency modulation depth.

### **Noise LPF/HPF/BPF**

Switches between lowpass, highpass and bandpass noise filters. Lowpass and bandpass are suitable (with high resonance) for effects such as filter snaps, while highpass is suitable for sounds such as high hats.

### **Sine 1 Level**

Specifies the relative level of Sine 1. Moving the control beyond the center position can result in distortion.

### **Sine 2 Level**

Specifies the relative level of Sine 2. Moving the control beyond the center position can result in distortion.

### **Noise Level**

Specifies the relative level of Noise. Moving the control beyond the center position can result in distortion.

## Connections

### MIDI In

Input for a MIDI signal, as from the MIDI Out of the MVC. You can also connect the MDS8 sequencer here, which was specifically designed for use with the drum oscillators.

### Gate

Input for a gate signal, as from an MVC or the gate of a sequencer.

### MIn (Mute In)

Connect the Mute Out of another drum synth or DVC (Drum Voice Control, see later) to this input. The Mute Outs of drum synths or DVCs transmit a message each time they are triggered. When the signal is received at Mute In, the current sound's envelope is cut off. This is useful to close an open hi-hat, for example. By connecting several outs to a Mute Adder (see later) mute groups can be implemented.

### MOut (Mute Out)

Connect this output to the Mute In of another drum synth or DVC. Mute Out transmits a signal whenever the drum synth is triggered.

### Out

The oscillator's audio output. You can connect this to one of the four Modular patch outputs.

**Tip: Use the Patch outputs 1-4 as independent outputs when using several drum oscillators.**

## Drum Oscillator

The Drum Oscillator is comprised of two sections: Sine and Noise. Sine produces the tonal component of the percussion instrument, while Noise provides the noise. With the two components, the Drum Oscillator is well suited to creating snare drum sounds. Electronic bass drums, with a noise attack, are also easy to create. An integrated Preset list lets you save sounds independent of the patches.

### Controls

#### Presets

Opens the oscillator's Preset list.

**Note:** Presets can play back at substantially high volumes. Therefore use a moderate volume level when auditioning them. Otherwise damage to your audio gear and/or your hearing could occur.



#### Note No.

Assigns the MIDI note number to which the oscillator will respond.

**Note:** This setting is not saved with a preset, but only with the patch. This way when you develop a drum map with several oscillators, and you want to change sounds, the key assignments remain appropriate for the song.

#### FVel

Assigns a velocity value to use when the sound is triggered by a gate signal.

**This value is effective only when the DrumSynth is triggered by Gate. When it is triggered via MIDI, the MIDI note-on velocity is used instead.**

#### Sine Level

Specifies the relative level of sine component. Moving the control beyond the center position can result in distortion.

#### Sine Decay

Sets the time it takes for sine component to fade to silence after being triggered.

#### Sine 809

Adjusts smoothly from a normal decay envelope to a hold-decay envelope. Turning the control to the right increases the 'hold' part of the envelope while reducing the 'decay' portion. The envelope time remains unchanged.

## NoiseL

Sets the volume level of the noise component. Moving the control beyond the center position can result in distortion.

## Noise Color

Adjusts the basic tone color of the noise component of your percussion instrument.

## Noise Decay

Sets the time it takes for noise component to fade to silence.

## Noise Slope

Adjusts the decay characteristic of the noise component. When set to the left, the decay is exponential (slower fadeout). When set to the right, the fade is exponential (relatively quick). Percussive sounds naturally fade exponentially, so set the control towards the right for percussive sounds.

## Snap

Adds an additional 'click' to the attack to further define a percussive 'strike'..

## TTune

Adjusts the basic pitch of the instrument.

## PMod

Adjusts the pitch modulation depth.

## PDec

Sets the duration of the pitch modulation. Depending on the PMod and PDec settings, you can add character to the sound anywhere from 'boomy' to 'scratchy'.

## Connections

### MIDI In

Input for a MIDI signal, as from the MIDI Out of the MVC. You can also connect the MDS8 sequencer here, which was specifically designed for use with the drum oscillators.

### Gate

Input for a gate signal, as from an MVC or the gate of a sequencer.

## Out

The oscillator's audio output. You can connect this to one of the four Modular patch outputs.

**Tip: Use the Patch outputs 1-4 as independent outputs when using several drum oscillators.**

### MIn (Mute In)

Connect the Mute Out of another drum synth or DVC (Drum Voice Control, see later) to this input. The Mute Outs of drum synths or DVCs transmit a message each time they are triggered. When the signal is received at Mute In, the current sound's envelope is cut off. This is useful to close an open hi-hat, for example. By connecting several outs to a Mute Adder (see later) mute groups can be implemented.

### MOut (Mute Out)

Connect this output to the Mute In of another drum synth or DVC. Mute Out transmits a signal whenever the drum synth is triggered.

## Percussion Oscillator

The Percussion Oscillator is the most basic of the drum oscillators. It includes only a single sine section with the typical drum parameters available. The Drum oscillator is suitable for bass drum, tom tom, and electronic drums. As with the other oscillators, this module has its own integrated Preset list.

### Controls

#### Controls

#### Presets

Opens the oscillator's Preset list.

**Note:** Presets can play back at substantially high volumes. Therefore use a moderate volume level when auditioning them. Otherwise damage to your audio gear and/or your hearing could occur.

#### Note No.

Assigns the MIDI note number to which the oscillator will respond.

**Note:** This setting is not saved with a preset, but only with the patch. This way when you develop a drum map with several oscillators, and you want to change sounds, the key assignments remain appropriate for the song.



#### FVel

Assigns a velocity value to use when the sound is triggered by a gate signal.

#### Tune

Adjusts the basic pitch of the instrument.

#### Decay

Sets the time it takes for the instrument to fade to silence after being triggered.

#### PMod

Adjusts the pitch modulation depth.

#### PDec

Sets the duration of the pitch modulation. Depending on the PMod and PDec settings, you can add character to the sound anywhere from 'boomy' to 'scratchy'.



**809**

Adjusts smoothly from a normal decay envelope to a hold-decay envelope. Turning the control to the right increases the 'hold' part of the envelope while reducing the 'decay' portion. The envelope time remains unchanged.

### **Snap**

Adds an additional 'click' to the attack to further define a percussive 'strike'.

### **Output**

Controls the output level of the instrument. Moving the control past the mid-point may result in distortion.

## **Connections**

### **MIDI In**

Input for a MIDI signal, as from the MIDI Out of the MVC. You can also connect the MDS8 sequencer here, which was specifically designed for use with the drum oscillators.

### **Gate**

Input for a gate signal, as from an MVC or the gate of a sequencer.

Connect the Mute Out of another drum synth or DVC (Drum Voice Control, see later) to this input. The Mute Outs of drum synths or DVCs transmit a message each time they are triggered. When the signal is received at Mute In, the current sound's envelope is cut off. This is useful to close an open hi-hat, for example. By connecting several outs to a Mute Adder (see later) mute groups can be implemented.

### **MIn (Mute In)**

Connect the Mute Out of another drum synth or DVC (Drum Voice Control, see later) to this input. The Mute Outs of drum synths or DVCs transmit a message each time they are triggered. When the signal is received at Mute In, the current sound's envelope is cut off. This is useful to close an open hi-hat, for example. By connecting several outs to a Mute Adder (see later) mute groups can be implemented.

### **MOut (Mute Out)**

Connect this output to the Mute In of another drum synth or DVC. Mute Out transmits a signal whenever the drum synth is triggered.

### **Out**

The oscillator's audio output. You can connect this to one of the four Modular patch outputs.

**Tip: Use the Patch outputs 1-4 as independent outputs when using several drum oscillators.**



## Drum Voice Control

You need the DVC when you want to design sophisticated drum circuits in the Modular. It allows envelope generators and amplifiers to be used with the Hihat Source (see later) or other oscillators, and basically provides connections to control these modules. A typical use would be to include a few DVCs in a patch, along with a Hihat Source, to simulate the various hi-hat states - open, closed, half open, etc.

Connecting the DVC is similar to connecting the MVC within a patch. Using the DVC is not necessarily simple, so several example Presets (hihat.mdl etc.) have been included. After examining these, however, you should have no trouble making creative use of the DVC.

### Controls

#### Note No.

Assigns a MIDI note number to the MVC.

#### Vel

Assigns a default velocity value to use when the sound is triggered by a gate signal.



### Connections

#### MIDI In

Input for a MIDI signal, as from the MIDI Out of the MVC. You can also connect the MDS8 sequencer here, which was specifically designed for use with the drum oscillators.

#### Gate

Input for a gate signal, as from an MVC or the gate of a sequencer.

#### Vel

Output for the MIDI velocity value. This can be either the value that arrived with the MIDI data, or the value set by the Vel control to be sent when a gate signal is received.

#### MIn (Mute In)

Connect the Mute Out of another drum synth or DVC (Drum Voice Control, see later) to this input. The Mute Outs of drum synths or DVCs transmit a message each time they are triggered. When the signal is received at Mute In, the current sound's envelope is cut off. This is useful to close an open hi-hat, for example. By connecting several outs to a Mute Adder (see later) mute groups can be implemented.

#### MOut (Mute Out)

Connect this output to the Mute In of another drum synth or DVC. Mute Out transmits a signal whenever the drum synth is triggered.

## Gate Out

The DVC sends a gate signal to trigger modules such as an envelope generator. The gate signal is output when a MIDI Note On is received, or when a gate signal is received at the DVC's Gate input.

## Esync

Feedback containing envelope status information. Connect this to the port of the same name on an envelope generator. If you are using several envelope generators, combine the Esync connections using the Esync Adder.

## Mute Adder 2

The Adder is required when you want to build groups of percussion instruments. This is the case when you want to connect more than to drum synths and/or DVCs. Two DVCs or drum synths can be connected together directly using their Mute Out and Mute In connections. With three drum synths/DVCs, however, you must use an adder. Connect two of the Mute Outs to the Ins in the Adder, and the Out of the Adder to the other synth or DVC to complete the group.

### Connections

#### In 1

Connect the Mute Out of the first drum synth or DVC you wish to merge into the group here.

#### In 2

Connect the Mute Out of the second drum synth or DVC you wish to merge into the group here.

#### Out

Provides a 'mixed' signal of the Mute Outs of the first two synths or DVCs. Connect this to the Mute In of the third synth/DVC.



## Hihat Source

The Hihat Source provides waveforms and spectra suitable for developing effective hi-hat sounds. Since this module uses several oscillators and a noise source to produce the spectra, and is thus resource hungry, it is best to load it only once. For other cymbal or hi-hat effects, use other modules from the Modular's module library. To illustrate the use of this module we have supplied an example patch, 'Hihat.mdl', along with the Modular tutorials. This also illustrates the use of the DVC to create groups. The HH Sound Source has its own Preset List, with which you can store the current settings.

### Controls

#### Tune 1-3

These controls alter the spectrum of the hi-hat waveform. Adjust these until you find a sound that is suitable for your purposes. Do not be surprised that adjusting these controls seems to be a chaotic procedure with rather unpredictable results - this is intended in order to create complex hi-hat and cymbal effects.

#### HPF

This High Pass Filter lies downstream from the oscillators controlled by Tune 1-3. If you find the spectrum is too pitched, or tonal, use this filter. Increase the value of the high pass filter until the tonality disappears.



#### Nz BPF (Noise BPF)

Nz BPF can be added to help achieve a metallic-sounding spectrum. This control filters the noise to 'tune' it to the overall sound. In other words, this control lets you adjust the color of the noise before adding it to the overall signal.

#### OSC Level

Controls the level of oscillators 1-3. This affects only the signal at the Out connection. Out is the Hihat Source's final mix output.

#### Noise Level

Controls the level of the noise component. This affects only the signal at the Out connection. Out is the Hihat Source's final mix output.

### Connections

#### Osc

The direct audio output of oscillators 1-3. Usually this is a very metallic sound, suitable for cymbal sounds etc.

#### Noise

The direct audio output of the noise component. This is useful for sounds such as electronic hi-hats.

#### Out

The audio mix output of the Hihat Source. The controls Osc Level and Noise Level affect the mix at this output.

## SampleDrum Oscillator

This is a monophonic sample-playback oscillator which is especially suited for use with drum and sound-effects samples. It can be assigned to a specific MIDI note number and played from a keyboard – it also works quite nicely with the MDS8 Drum Sequencer module. An integrated volume envelope permits tailoring of the playback of longer sounds, including sound cut-off in response to note-off events.

### Load

Press this button to load a sample, the standard dialog for loading a file appears.

**The oscillator can only load individual samples, not complete Akai programs.**

### Coarse

Permits coarse tuning of the sample in semitones.

Note that sample tuning is accomplished via alteration of the effective sample playback "speed".

### Fine

Permits fine-tuning of sample pitch.



### Vel

Controls the amount of influence of note velocity on sample loudness.

### Note

Specifies the MIDI note number to which the oscillator will respond.

### A

Sets the attack time. When the envelope generator receives a gate signal, it starts the attack phase, and the modulation signal rises to the maximum value in the time set here.

### H

Sets the Hold time. When the attack phase has completed, the level of the envelope signal is held at maximum for the duration set here.

### D

Sets the decay time. After the hold phase has finished, the modulation signal falls to zero according to the time and characteristic settings.

## ASlope

Adjusts the slope curve for the attack phase. The curve is continuously adjustable from linear to exponential.

## DSlope

Adjusts the slope curve for the decay phases. The curve is continuously adjustable from a linear to a logarithmic fade out.

## Connections

### MIDI

Input for a MIDI signal, as from the MIDI Out of the MVC. You can also connect the MDS8 sequencer here, which was specifically designed for use with the drum oscillators.

### MOut (Mute Out)

When this output is connected to the Mute In jack of another Drum Oscillator, triggering of sample playback will cause playback in the connected oscillator to be immediately stopped.

### MIn (Mute In)

When this input is connected to the Mute Out jack of another Drum Oscillator, sample playback will be immediately stopped whenever playback in the connected oscillator is triggered.

The Mute In/Out jacks of two SampleDrum oscillators can be cross-connected so that each oscillators stops the other. This would be a typical setup for playback of open and closed hi-hat samples.

### Out

The audio signal from the oscillator.

## Utilizing Samples

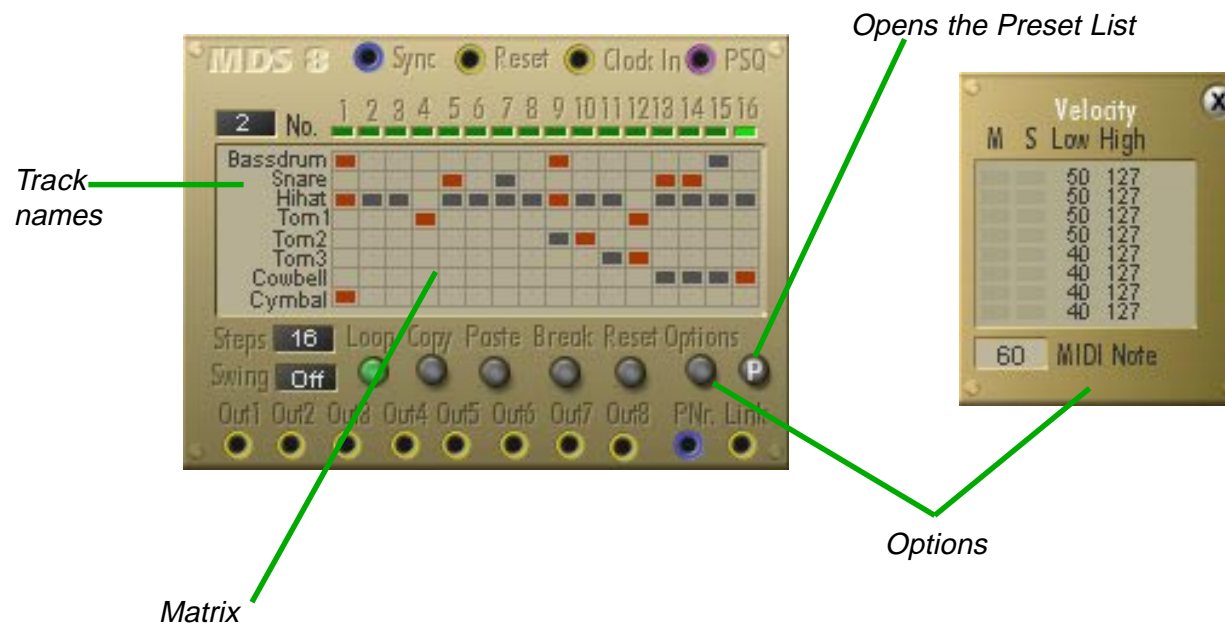
Once you've loaded a sample and then save a patch preset, the same sample will later be reloaded whenever you recall this preset. Keep in mind that the SampleDrum Oscillator does not store the sample itself within the preset, but merely records its name and its location. If you subsequently delete this sample, rename it or move it to a different location (or if the sample was loaded from a CD-ROM which is not inserted in the CD-ROM drive), the SampleDrum Oscillator will not be able to find it and therefore not be able to load it.

**In particular, note that when transferring a patch which uses samples to a different computer, it is also necessary to transfer the samples being used.**

To get around this problem, read the chapter OSC **in the section**, Sample Pool.

## MDS 8

The MDS8 (**MIDI Drum Sequencer**) is designed especially for use with the Drum Oscillator/Synthesizer modules. It has 8 independent tracks, each of which is assigned to its own MIDI output. With the MDS8, you can program complete drum patterns and transform your patch into a true groove machine. In connection with the pattern sequencer, you can program complete rhythm tracks, just as you're used to doing with hardware drum machines. Rhythms are programmed easily via the built-in pattern matrix, and the free accent per step permits the creation of highly dynamic rhythms.



## No.

Use this text fader to set the pattern to be played.

**Note: this control is disabled by a PSQ connection to the pattern sequencer.**

## Matrix

Here, you set the steps at which MIDI triggers are to occur.

## Navigating in the Matrix

Once you've clicked inside the matrix – e.g., to set a step – you can use the arrow keys on your computer keyboard to move around within it.

## Setting steps

There are three ways to set and clear steps:

1.) Click with the mouse on the step you wish to set. A second click on the same step sets an accent for the step, a third clears the step.

2.) Use the arrow keys to navigate through the matrix, and repeated strokes on the <Space> bar to set a step, set an accented step, or clear the step.

3.) Same as 2) above, but type 1 to clear a step, 2 to set it, and 3 to accent it.

## Steps

Sets the desired length for the pattern.

**When you are controlling the MDS8 remotely over PSQ connection using the Pattern Sequencer, this parameter can no longer be adjusted here.**

## Swing

Controls the amount of rhythmic "swing".

## Loop

Activating this option causes the current song or pattern-chain to repeat indefinitely.

**If you want to control the MDS 8 with the Pattern SEQ Loop must be *On*.**

## Copy

Click here to copy the current sequence into the clipboard.

## Paste

Click here to replace the current sequence with a sequence previously copied into the clipboard. This exchange is possible only within a single sequencer module.

You can also paste a copied sequence after switching to a new preset, which permits transfer of sequences from one preset to another.



## Break

Clicking on this button stops the sequencer.

## Reset

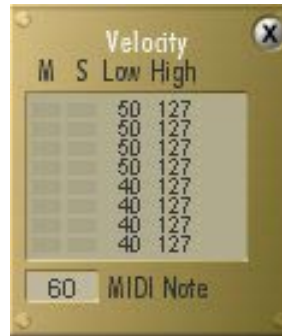
Click on **Reset** to restart the pattern from step 1. This can also be done from outside via the Reset input – permitting restart to be triggered by notes played on a keyboard, for example. (see also Gate SEQ)

## Preset

Click the P button to open the module's preset list. Each preset contains 32 independent sequences.

## Options

Click Options to open the following dialog:



## Mute

Click Mute to mute the corresponding track.

## Solo

Click Solo to hear only the corresponding track.

## Velocity Low

Sets the MIDI note velocity for normal steps.

## Velocity High

Sets the MIDI note velocity for accented steps.

## Connections

### Sync

Triggers advance to the next step in the pattern list. A typical connection would be to the Sync output of the MDS8 sequencer, which sends a sync signal upon reaching the end of its current pattern.

### Clock In

Connect the clock output of the MIDI Clock module here – or better still, the output of a Clock Divider.

### Reset

Connect this signal to other step sequencers in order to have them simultaneously reset when the pattern sequencer sends a stop signal.

### PSQ

For connection to the input of the same name on a step sequencer, so that its pattern length and number of steps can be remote-controlled by the pattern sequencer.

## **Out 1-8**

Connect these MIDI outputs to the MIDI inputs of Drum Oscillators or the Drum Synthesizer.

**Don't forget to set the MDS8 and the Drum Module to the same MIDI note number.**

## **PNr.**

Connect the PS32 pattern switcher here, for example, to permit live selection of patterns from the keyboard. Note that this is possible only when the PSQ input is not connected.

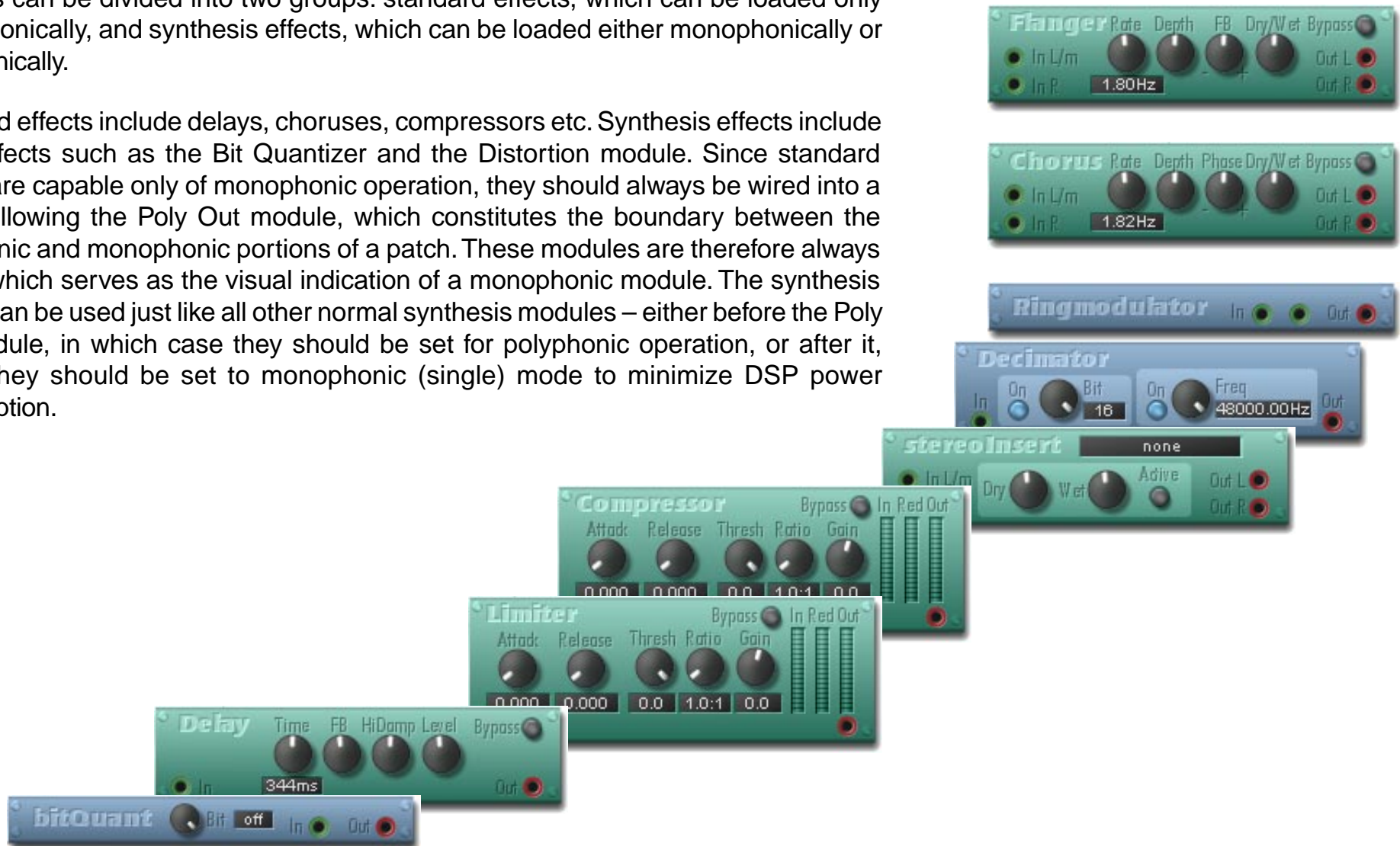
## **Link**

Use this signal to supply clocks to additional sequencer modules. Note that swing settings may affect this signal.

# Effects

The SCOPE Fusion Platform provides a complete assortment of effects modules. The modules can be divided into two groups: standard effects, which can be loaded only monophonically, and synthesis effects, which can be loaded either monophonically or polyphonically.

Standard effects include delays, choruses, compressors etc. Synthesis effects include other effects such as the Bit Quantizer and the Distortion module. Since standard effects are capable only of monophonic operation, they should always be wired into a patch following the Poly Out module, which constitutes the boundary between the polyphonic and monophonic portions of a patch. These modules are therefore always green, which serves as the visual indication of a monophonic module. The synthesis effects can be used just like all other normal synthesis modules – either before the Poly Out module, in which case they should be set for polyphonic operation, or after it, where they should be set to monophonic (single) mode to minimize DSP power consumption.



# Compressor

Standard effect, always monophonic

This effect modifies the dynamics of a sound. The level of louder passages is decreased. This means that the overall level of the sound can be set higher, with the net result that the level of softer sections is increased. The compressor operates by monitoring the level of the input signal. An adjustable threshold level determines the point at which compression begins. Attack and release controls determine how quickly the compressor responds when the threshold is exceeded and when the input signal level falls back below the threshold, respectively. The ratio control determines the intensity of the compression – i.e., the ration of input level change to output level change. The gain control adjusts the level of the compressed signal.

## Controls

### Attack

The Attack time (in milleseconds) is the compressor's reaction time - the time it takes it to respond to a level spike.

### Release

This is the time (in milliseconds) after the signal falls back under the threshold that compression is no longer active.

### Threshold

Sets the input signal level above which compression begins.



### Ratio

The Ratio adjusts the compression rate for signals that exceed the threshold level. The compression is displayed as a relation value. 1:1 means that there is no compression. 3:1e.g. means that a an amplification of +3dB of the input signal results only in +1dB at the output.

### Gain

A compressor reduces the transient levels of a signal so that the overall signal can be increased later without danger of distortion. This increases the average level of the music, resulting in a fuller sound. Adjust the volume increase with the Gain control.

### Bypass

Sends the input signal directly to the output, bypassing the effect.

## Connections

### In

The audio input of the compressor.

### Out

The audio output of the compressor .

# Limiter

Standard effect, always monophonic

This effect is related to compression and likewise modifies the dynamics of a sound. The level of louder passages is decreased. This means that the overall level of the sound can be set higher, with the net result that the level of softer sections is increased. The limiter operates by monitoring the level of the input signal. An adjustable threshold level determines the point at which limiting begins. Attack and release controls determine how quickly the limiter responds when the threshold is exceeded and when the input signal level falls back below the threshold, respectively. The ratio control determines the intensity of the limiting – i.e., the ration of input level change to output level change. The gain control adjusts the level of the limited signal.

## Controls

### Attack

The Attack time (in milleseconds) is the limiter's reaction time - the time it takes it to respond to a level spike.

### Release

This is the time (in milliseconds) after the signal falls back under the threshold that limiting is no longer active.

### Threshold

Sets the input signal level above which limiting begins.



### Ratio

The Ratio adjusts the limiting rate for signals that exceed the threshold level. The limiting is displayed as a relation value. 1:1 means that there is no compression. 3:1e.g. means that a an amplification of +3dB of the input signal results only in +1dB at the output.

### Gain

A limiter reduces the transient levels of a signal so that the overall signal can be increased later without danger of distortion. This increases the average level of the music, resulting in a fuller sound. Adjust the volume increase with the Gain control.

### Bypass

Sends the input signal directly to the output, bypassing the effect.

## Connections

### In

The audio input of the limiter.

### Out

The audio output of the limiter .

# Delay

Standard effect, always monophonic

The Delay or Echo delays the signal in simple or more complex ways producing individual or repeated echos. Repeated echos are created using a feedback. The feedback chain contains a filter which allows a damping of high frequencies.

## Controls

### Time

Adjusts the delay time in milliseconds.

### FB

Sets the strength of the feedback - the portion of the delayed signal that is routed back to the input to create multiple receding echos. The degree of feedback determines the fade out time of the repeated echos.

### HiDamp

Controls the amount of high frequency damping in the feedback signal. Damping degrades the signal with each repeat, creating a natural sounding fade.



### Dry

The level of the original signal.

### Wet

The level of the effect signal.

### Bypass

Sends the input signal directly to the output, bypassing the effect.

## Connections

### In

The audio input of the delay.

### Out

The audio output of the delay .

# Tempo Delay

Standard effect, always monophonic

This module is actually a Stereo Delay with optional cross feedback. The reason the word 'tempo' appears in its name is to signify that it can be synchronized to the MIDI clock.

## Controls

### TimeL/R

Adjusts the delay time, in milliseconds, for the left and/or right channels.

### External

When this is enabled (the button is lit) you can control the Delay times externally. Connect the MIDI clock through a Frequency Divider to the FL and FR inputs. The delay times are determined by the Tempo and and settings in the Frequency Divider.

### FB

Controls how much of the delayed signal is fed back to the inputs to be delayed once again.



### Cross

This switch changes the feedback loop to a Cross Feedback loop: The output of the left channel is fed back to the right channel input, and the right channel output is routed to the left input channel.

### HiDamp

Adjusts the amount of high frequency filtering applied to each cycle of the signal in the feedback loop.

### Dry

Controls the level of the original signal in the output.

### Wet

Controls the level of the delayed signal in the output.

### Bypass

Routes the signal directly from the input to the output, bypassing the effect.

## Connections

### In L/m

Input for the left channel audio signal. This is also the input for a mono signal.

### In R

Input for the right channel audio signal. When you connect a signal here, the effect switches automatically to stereo mode.

### FL (Frequency Left)

Input for a MIDI Clock, or, better yet, a Frequency Divider to control the effect's left channel delay time.

### FR (Frequency Right)

Input for a MIDI Clock, or, better yet, a Frequency Divider to control the effect's right channel delay time.

### Out L

Left channel output.

### Out R

Right channel output.



# Chorus

Standard effect, always monophonic

The name „chorus“ hints at the sound produced by this effect. It spreads and thickens the sound passed through it, simulating the sound of multiple instruments of the same type playing together – in other words, a chorus. This effect is achieved by means of a short delay line whose delay time is periodically modulated. Mixing of this delayed signal with the original produces the chorus effect. The intensity of the effect depends upon the modulation rate, depth and phase settings as well as the dry/wet (original/ delayed) mix. The effect is also useful for creating a stereo sound from a monaural signal.

## Controls

### Rate

Controls the frequency of the pitch modulation.

### Depth

Adjusts the modulation depth - the strength of the pitch modulation.

### Phase

Adjusts the phase difference of the modulation signal between the left and right channels. This influences the ‘width’ of the stereo field.



### Dry/Wet

Determines the relation between the level of the original and the effect signal.

### Bypass

Sends the input signal directly to the output, bypassing the effect.

## Connections

### In L/m

The left audio input. This input also is used for monophonic signals

### In R

The right audio input. The Chorus is automatically switched to stereo input mode once a signal is connected to this jack.

### Out L

The left channel of the audio output.

### Out R

The right channel of the audio output.

# Flanger

Standard effect, always monophonic

This effect is similar to the chorus. Like the chorus, a flanger is based on a delay line whose delay time is periodically modulated. However, the delay times in a flanger are substantially shorter than those of a chorus. In addition, the flanger utilizes feedback of the delayed signal back to the delay line input. Therefore, it not only thickens the sound but can add noticeable coloration owing to the comb-filter effect which results from the feedback. The intensity of the effect depends upon the modulation rate, depth and phase settings as well as the dry/wet (original/delayed) mix. The effect is also useful for creating a stereo sound from a monaural signal.

## Controls

### Rate

Controls the frequency of the delay modulation of the flanger.

### Depth

Adjusts the modulation depth - the strength of the delay modulation.

### FB

Adjusts the feedback depth, or the amount of processed signal fed back into the input.



### Dry/Wet

Determines the relation between the level of the original and the effect signal.

### Bypass

Sends the input signal directly to the output, bypassing the effect.

## Connections

### In L/m

The left audio input. This input also is used for monophonic signals

### In R

The right audio input. The Flanger is automatically switched to stereo input mode once a signal is connected to this jack.

### Out L

The left channel of the audio output.

### Out R

The right channel of the audio output.

## Tempo Flanger

Standard effect, always monophonic

This is a Flanger module in which the modulation speed can be synchronized to a MIDI Clock. For information on what a flanger does, and how to control it, read the previous section. This section explains only the added parameters and connections.

### Additional Parameter

#### External

By enabling this option (button is lit) you can control the modulation speed externally. Connect a MIDI Clock through a Frequency Divider to the Ext Freq input. The modulation speed is now derived from the tempo, and the settings in the Frequency Divider. The Rate button is disabled when in external mode.



### Additional Connections

#### Ext Freq

Connection for a frequency signal, or, better, a signal from a Frequency Divider, to set the modulation rate.

# Phaser

Standard effect, always monophonic

The phaser or phase-shifter is similar to the flanger and chorus, but instead of using a delay line, it functions by introducing frequency-dependent phase shifts into the signal which is passed through it. The key phase-shift frequencies, and therefore the amount of phase shift, are periodically modulated. The phase-shifter effect results when this phase-manipulated signal is mixed with the original signal. The effect is thus similar to the flanger and chorus, but with a sound character of its own. The intensity of the effect depends upon the modulation rate, depth and phase settings as well as the dry/wet (original/phase-shifted) mix. The effect is also useful for creating a stereo sound from a monaural signal.

## Controls

### Rate

Controls the frequency of the phase modulation of the Phaser.

### Depth

Adjusts the modulation depth - the strength of the phase modulation.

### FB

Adjusts the feedback depth, or the amount of processed signal fed back into the input.



### Dry/Wet

Determines the relation between the level of the original and the effect signal.

### Bypass

Sends the input signal directly to the output, bypassing the effect.

## Connections

### In L/m

The left audio input. This input also is used for monophonic signals

### In R

The right audio input. The Phaser is automatically switched to stereo input mode once a signal is connected to this jack.

### Out L

The left channel of the audio output.

### Out R

The right channel of the audio output.

## Tempo Phaser

Standard effect, always monophonic

This is a Phaser module in which the modulation speed can be synchronized to a MIDI Clock. For information on what a phaser does, and how to control it, read the previous section. This section explains only the added parameters and connections.

### Additional Parameter

#### External

By enabling this option (button is lit) you can control the modulation speed externally. Connect a MIDI Clock through a Frequency Divider to the Ext Freq input. The modulation speed is now derived from the tempo, and the settings in the Frequency Divider. The Rate button is disabled when in external mode.



### Additional Connections

#### Ext Freq

Connection for a frequency signal, or, better, a signal from a Frequency Divider, to set the modulation rate.

# Distortion

Synthese-Effekt, mono- und polyphon

This effect produces both „soft“ and „hard“ distortion. Soft distortion is comparable to that which results when analog magnetic tape overload occurs – it comes on slowly with increasing signal level and has a soft sound. Hard distortion is produced via simple clipping. It appears more suddenly and has a harder, edgier sound. This module can be used polyphonically as well as monophonically – thus both before and after the Poly Out module.

## Controls

### Dist

Controls the intensity of the distortion for both the Soft and Hard output.



## Connections

### In

Input for audio signals.

### Soft

Output for the distorted signal.

### Hard

Output for the strongly distorted signal.

## Bit Quantizer

Synthesis effect, mono- und polyphonic

In the SCOPE Fusion Platform, signals are normally computed with 32-bit precision. The Bit Quantizer lets you deliberately and radically decrease this precision to the point where noticeable quantization artifacts begin to appear – the characteristic „digital trash“ sound and limited dynamic range of quantization noise. This module can be used polyphonically as well as monophonically – thus both before and after the Poly Out module.



### Controls

#### Bit

Controls the number of bits which is used for quantization.

### Connections

#### In

Input for audio signals.

#### Out

Output for the quantized signal.

Modular

## Decimator

Synthesis effect, mono- und polyphonic

The Decimator lets you play a signal with a different bit resolution and sample rate as the rest of the system. In effect, the signal is internally resampled. Aliasing and quantization noise can be deliberately produced, depending upon the settings used. This module can be used polyphonically as well as monophonically – thus both before and after the Poly Out module.

### Controls

#### Bit

Controls the number of bits which is used for the Bit reduction.

#### Bit on/off

Turns on/off the bit reduction.

#### Sample Rate

Controls the sample rate which is used for the conversion.

#### FmA

Here, the sample rate can be modulated – the potentiometer controls the intensity of modulation.

#### Sample Rate on/off

Turns on/off the sample rate conversion.



### Connections

#### In

Input for audio signals.

#### Fmod

Input for modulation signals.

#### Out

Output for the quantized signal.

Contents

Index

127

# Ringmodulator

Synthese-Effekt, mono- und polyphon

The ring-modulation effect is often described as „bell-like“ or „inharmonic-sounding“ – fitting descriptions, when, for example, two audio oscillators are passed through the ring modulator. In essence, however, the ring modulator simply outputs the (multiplication) product of its two input signals, and therefore is also useful for modulation of modulation signals. This module can be used polyphonically as well as monophonically – thus both before and after the Poly Out module. As the ring modulator offers only connections and no controls of its own, the intensity of the effect must be controlled from outside, e.g., using mix modules.

## Connections



### In1

Input for audio and modulation signals.

### In2

Input for audio and modulation signals.

### Out

Output of the modulated signals.



## Mono/Stereo Insert

With these modules you can integrate any SCOPE Fusion Platform effects into your Patch.



Please take note of the following: When you exchange Modular patches with other users, it is possible that you or the other user may not have a copy of an effect offered by a third-party provider. The effect slot of the insert module will in that event remain empty – either it should be switched off, or alternatively, one of the SCOPE Fusion Platform effects should be loaded.

### Controls

#### Insert Slot

Load any SCOPE Fusion Platform effect into this slot. By double-clicking on the name in this slot you open the effect's user surface. To remove the effect, select the textfield in the slot and press the <Delete> or <NumLock> key on your computer keyboard.

#### Dry

Controls the level of the original signal.



#### Wet

Controls the level of the effect signal.

Note that the Dry and Wet settings apply only to the Insert module itself. The individual effects modules likewise possess Dry and Wet controls of their own, which must be set appropriately.

#### Active

This button activates the effect in the insert slot, which is active when the button is illuminated. When the insert slot is deactivated, the effect is also unloaded from the DSPs and the audio input signal to the effect is passed directly to the output.

If the insert slot is activated with no effect loaded, the signal in the insert module is determined by the Dry level control. If this is set to minimum, no signal will come through.

## Connections

### Mono Insert

#### In

Input for audio signals.

#### Out

Output of the audio signals.

### Stereo Insert

#### InL/m

Input for the left audio signal or a monophonic signal.

#### InR

Input for the right audio signal.

#### Out L

Output of the left audio signal.

#### Out R

Output of the right audio signal.

# Sequencer Modules

The Modular2 offers a full assortment of sequencer modules which can be used to produce rhythmic structures, note runs, filter sweeps, drum patterns and much more. The various types of sequencer modules produce different types of output signal as appropriate. For example, the pitch sequencer produces pitch control signals which can be used to control an oscillator in semitone increments, the gate sequencer produces signals for triggering of envelopes, and the various control sequencers permit sequence-driven control of filter, pan and other modules.

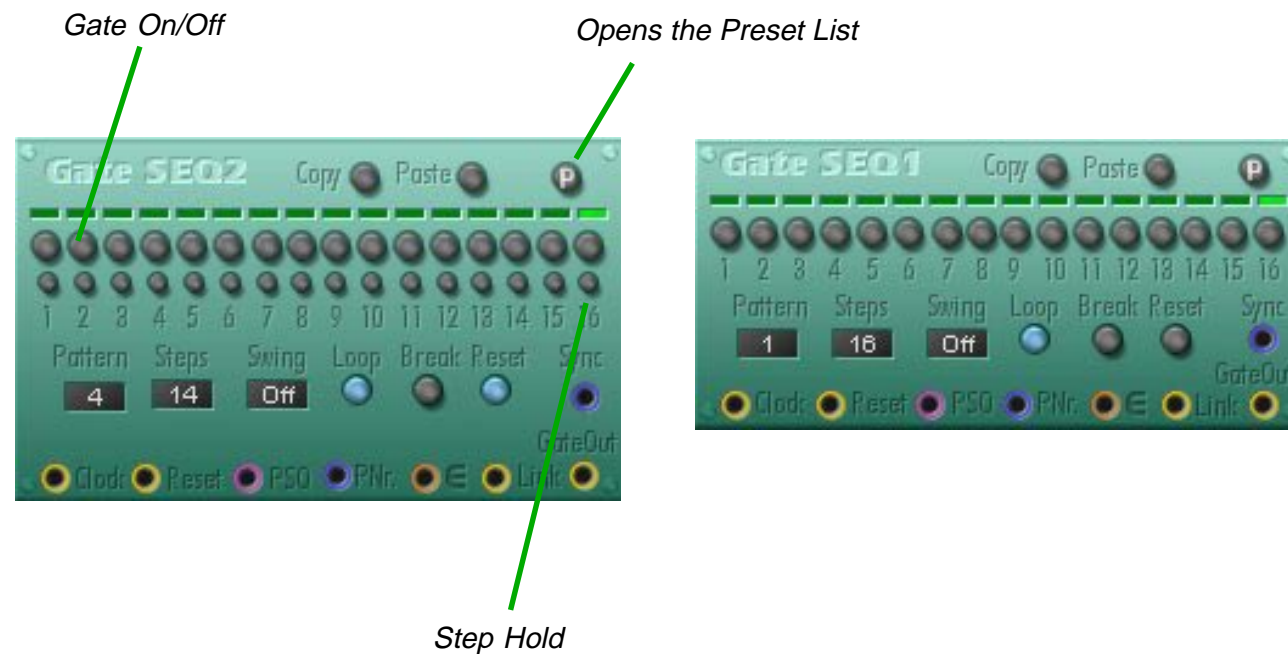
The pattern switcher lets you recall patterns from a keyboard, while the pattern sequencer permits the creation of complete song structures.

These modules have some special-purpose connections. For example, the PSQ jack is provided especially for communication with the pattern sequencer.

The control sequencer modules are currently restricted to monophonic operation. This means that they can modulate polyphonic sounds, but that the modulation itself is the same for all voices. For example, if the pitch sequencer is sending an offset of two semitones as you play a three-note chord, all three notes will be shifted in pitch by two semitones. However, polyphonic control sequencers are planned.

## Gate SEQ1 / SEQ2

The gate sequencers are used primarily for triggering envelopes. They output not only values indicating On and Off, but by means of Esync communication can also notify an envelope that it should "ramp down" to zero before being retriggered. SEQ1 puts out a trigger signal on every 16th-note time interval, while SEQ2 permits note lengths to be extended.



## Copy

Click here to copy the current sequence into the clipboard.

## Paste

Click here to replace the current sequence with a sequence previously copied into the clipboard. This exchange is possible only within a single sequencer module.

You can also paste a copied sequence after switching to a new preset, which permits transfer of sequences from one preset to another.

## Preset

Click the P button to open the module's preset list. Each preset contains 32 independent sequences.

## Gate On/Off

These sixteen buttons determine whether or not a trigger signal will be sent in the corresponding time interval. The **Step Hold** buttons extend note length by suppressing new triggers (see following).

### Step Hold (Gate SEQ2 only)

These buttons can be used to extend a note over one or more steps. Normally, a note-off is sent halfway through each step. If Step Hold is switched on for a step, then no note-off occurs in that step, and no note-on occurs at the start of the next step, with the result that the original note is stretched.

## Pattern

Here, you specify which of the 32 available patterns is to be played. This can also be controlled externally via the PNr input. Use the PS32 pattern switcher module to select patterns from a keyboard.

## Steps

Pattern length is independently variable for each pattern, up to a maximum of 16 steps.

**If the sequencer module is being remote-controlled by the Pattern Sequencer, the Pattern and Steps settings are ignored – in this case, the Pattern Sequencer determines which pattern will be played as well as the pattern length.**

## Swing

Controls the amount of rhythmic "swing".

This setting applies in common to all of a module's patterns.

## Loop

Activate this function to cause the pattern to repeat continuously.

Note that this option must be active when the sequencer module is to be remote-controlled by the Pattern Sequencer.

## Break

Clicking on this button stops the sequencer.

## Reset

Click on **Reset** to restart the pattern from step 1. This can also be done from outside via the Reset input – permitting restart to be triggered by notes played on a keyboard, for example.

**If you want to trigger Reset from the keyboard, connect the GateOr module to the Gate output of the MVC, and connect the output of the GateOr module to the Reset input of the Gate SEQ module. With this connection, Reset will be triggered on every new note. If the connection is made directly rather than via the GateOr module, Reset will trigger less often – with a polyphony of four, for example, only on every fourth note.**

## Connections

### Clock

Connect the clock output of the MIDI Clock module here – or better still, the output of a Clock Divider.

### Reset

Connect a trigger signal here to permit remote-actuated restarting of a pattern from step 1.

### PSQ

For connection to the signal output of the same name on the Pattern Sequencer, which can then remote-control pattern number and pattern length.

### PNo.

Connect the PS32 Pattern Switcher here to permit selection of patterns from the keyboard.

### E

Connect the Esync output of an ADSR envelope here. With this connection, the GateSequencer can command full "ramp down" of the envelope before retriggering it.

## Link

Here, connect other sequencer modules which are to be strongly coupled to the Gate SEQ module. A "swung" clock may also be output via Link.

**For example, if you use the OnGate option in a pitch sequencer, you need the clock signal from the Link output in order to relay information regarding whether or not a given step is set.**

## Gate Out

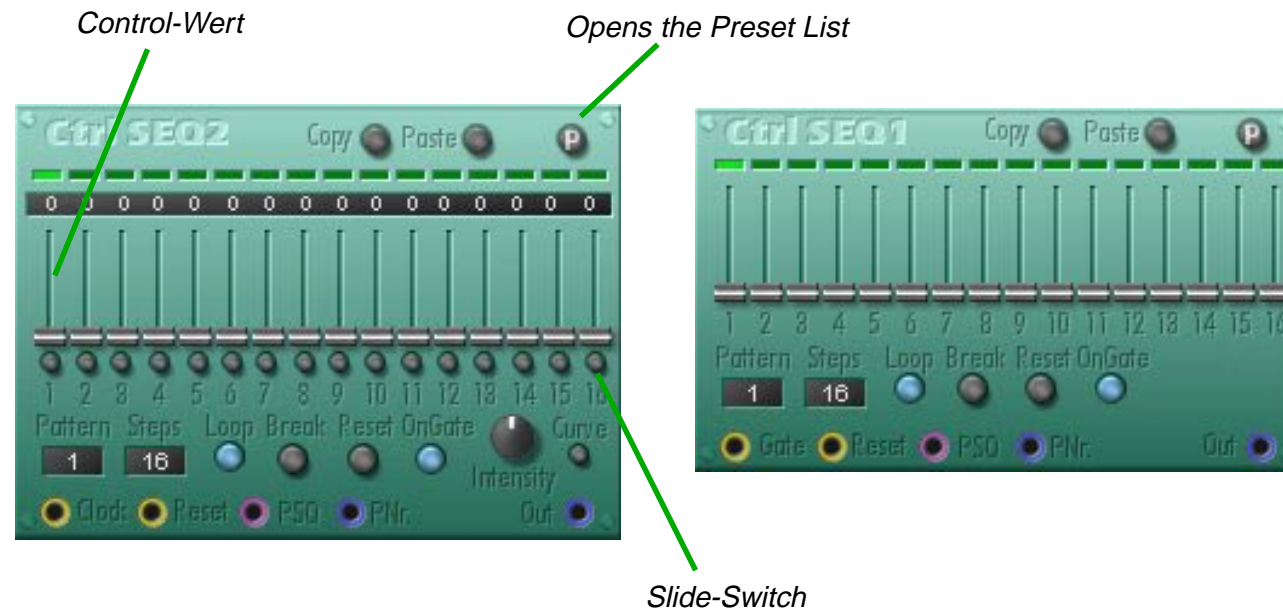
Here, connect an envelope or other module which is to be triggered on every step.

## Sync

Connect this output to the Sync input of the Pattern Sequencer, which thus receives a trigger signal when the end of the current pattern is reached and in turn can call up the next pattern.

## Ctrl SEQ1 / SEQ2

The control sequencers are used for control of modulation signals. They produce sequences of unipolar values in the range 0-127. Unipolar means that the modulation works in only one direction – for example, the preset cutoff of a filter can only be raised. At your option, the control value programmed for each step can either always be sent or sent only when a gate occurs on that step. Finally, the slide buttons permit step-by-step control over the change from one control value to the next, which can be either an instantaneous change or a smooth glide.



## Copy

Click here to copy the current sequence into the clipboard.

## Paste

Click here to replace the current sequence with a sequence previously copied into the clipboard. This exchange is possible only within a single sequencer module.

You can also paste a copied sequence after switching to a new preset, which permits transfer of sequences from one preset to another.

## Preset

Click the P button to open the module's preset list. Each preset contains 32 independent sequences.

## Control Value

Use these faders to set control values for each step in a pattern (or enter values in the range 0-127 directly via the text field above each fader).

## Slide (Ctrl SEQ2 only)

These three-state buttons permit you to activate a control-value glide function on a step-by-step basis. The glide can be either exponential (blue) or logarithmic (yellow). Which type is to be preferred depends upon the parameter being modulated – for example, filters often sound better with exponential control.

Depending upon the tempo and your own personal taste, you may also need to adjust the Intensity control (see below) to get the glide function to sound the way you want.

## Intensity (Ctrl SEQ2 only)

Controls the "steepness" or speed of the exponential or logarithmic control-value glide in pattern steps for which the Slide function is active (all steps are affected in common).

## Curve (Ctrl SEQ2 only)

Same as the Slide buttons described above – but switches the states of these buttons in common for all sixteen pattern steps at one time.



## Pattern

Here, you specify which of the 32 available patterns is to be played. (see also Gate SEQ)

## Steps

Pattern length is independently variable for each pattern, up to a maximum of 16 steps. (see also Gate SEQ)

## Swing

Controls the amount of rhythmic "swing". (see also Gate SEQ)

## Loop

Activate this function to cause the pattern to repeat continuously. (see also Gate SEQ)

## Break

Clicking on this button stops the sequencer.

## Reset

Click on **Reset** to restart the pattern from step 1. This can also be done from outside via the Reset input – permitting restart to be triggered by notes played on a keyboard, for example. (see also Gate SEQ)

## OnGate

Activate this option to cause control values to be sent only on steps in which a gate event (note-on) occurs. For this to work, the Ctrl SEQ Clock must come from the Gate SEQ module and not directly from the MIDI Clock module or Clock Divider module.

When this option is *not* active, control values are sent on every step, regardless of whether a note-on occurs, meaning that new control values may change the sound of an already-playing note from an earlier step.

## Connections

### Clock

Connect the clock output of the MIDI Clock module here – or better still, the output of a Clock Divider.

### Reset

Connect a trigger signal here to permit remote-actuated restarting of a pattern from step 1.

### PSQ

For connection to the signal output of the same name on the Pattern Sequencer, which can then remote-control pattern number and pattern length.

### PNo.

Connect the PS32 Pattern Switcher here to permit selection of patterns from the keyboard.

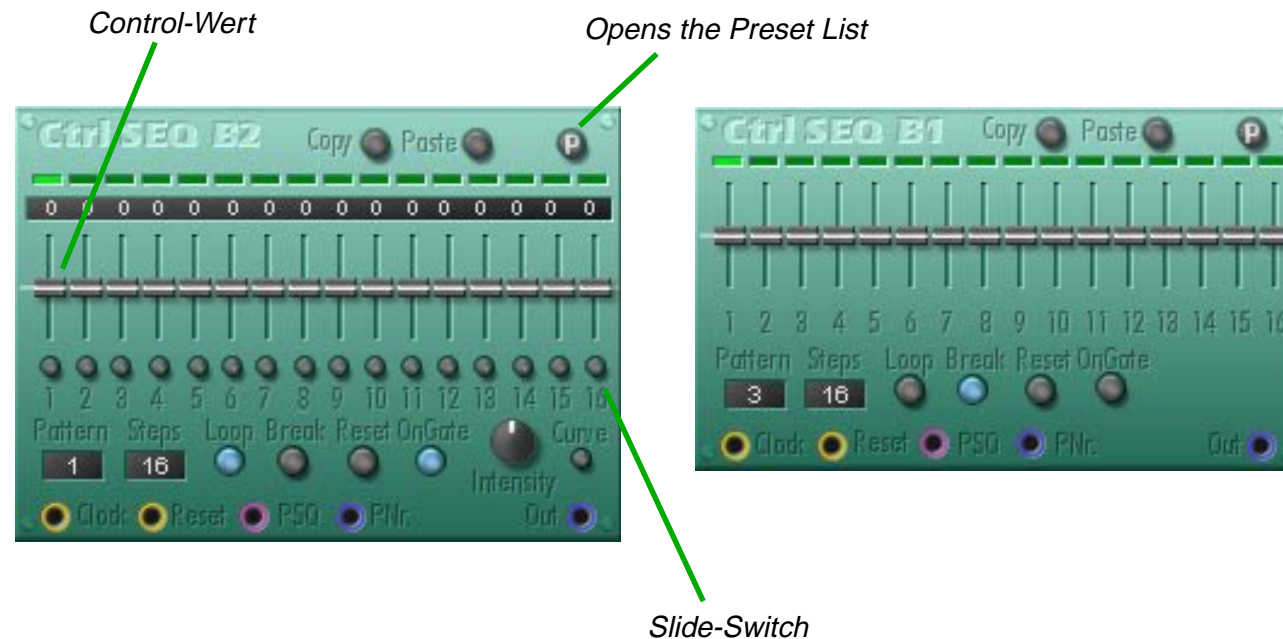
### Out

The unipolar modulation signal is output here.



## Ctrl SEQ B1 / SEQ B2

The "B" control sequencers are used for bipolar control of modulation signals. They produce sequences of control values in the range -64 - +63. Bipolar means that the modulation can work in both directions – for example, the preset cutoff of a filter can be both raised and lowered. At your option, the control value programmed for each step can either always be sent or sent only when a gate occurs on that step. Finally, the slide buttons permit step-by-step control over the change from one control value to the next, which can be either an instantaneous change or a smooth glide



## Copy

Click here to copy the current sequence into the clipboard.

## Paste

Click here to replace the current sequence with a sequence previously copied into the clipboard. This exchange is possible only within a single sequencer module.

You can also paste a copied sequence after switching to a new preset, which permits transfer of sequences from one preset to another.

## Preset

Click the P button to open the module's preset list. Each preset contains 32 independent sequences.

## Control Value

Use these faders to set control values for each step in a pattern (or enter values in the range -63 to +63 directly via the text field above each fader).

## Slide (Ctrl SEQ B2 only)

These three-state buttons permit you to activate a control-value glide function on a step-by-step basis. The glide can be either exponential (blue) or logarithmic (yellow). Which type is to be preferred depends upon the parameter being modulated – for example, filters often sound better with exponential control.

Depending upon the tempo and your own personal taste, you may also need to adjust the Intensity control (see below) to get the glide function to sound the way you want.

## Intensity (Ctrl SEQ B2 only)

Controls the "steepness" or speed of the exponential or logarithmic control-value glide in pattern steps for which the Slide function is active (all steps are affected in common).

## Curve (Ctrl SEQ B2 only)

Same as the Slide buttons described above – but switches the states of these buttons in common for all sixteen pattern steps at one time.

## Pattern

Here, you specify which of the 32 available patterns is to be played. (see also Gate SEQ)

## Steps

Pattern length is independently variable for each pattern, up to a maximum of 16 steps. (see also Gate SEQ)

## Swing

Controls the amount of rhythmic "swing". (see also Gate SEQ)

## Loop

Activate this function to cause the pattern to repeat continuously. (see also Gate SEQ)

## Break

Clicking on this button stops the sequencer.

## Reset

Click on **Reset** to restart the pattern from step 1. This can also be done from outside via the Reset input – permitting restart to be triggered by notes played on a keyboard, for example. (see also Gate SEQ)

## OnGate

Activate this option to cause control values to be sent only on steps in which a gate event (note-on) occurs. For this to work, the Ctrl SEQ Clock must come from the Gate SEQ module and not directly from the MIDI Clock module or Clock Divider module.

When this option is *not* active, control values are sent on every step, regardless of whether a note-on occurs, meaning that new control values may change the sound of an already-playing note from an earlier step.

## Connections

### Clock

Connect the clock output of the MIDI Clock module here – or better still, the output of a Clock Divider.

### Reset

Connect a trigger signal here to permit remote-actuated restarting of a pattern from step 1.

### PSQ

For connection to the signal output of the same name on the Pattern Sequencer, which can then remote-control pattern number and pattern length.

### PNo.

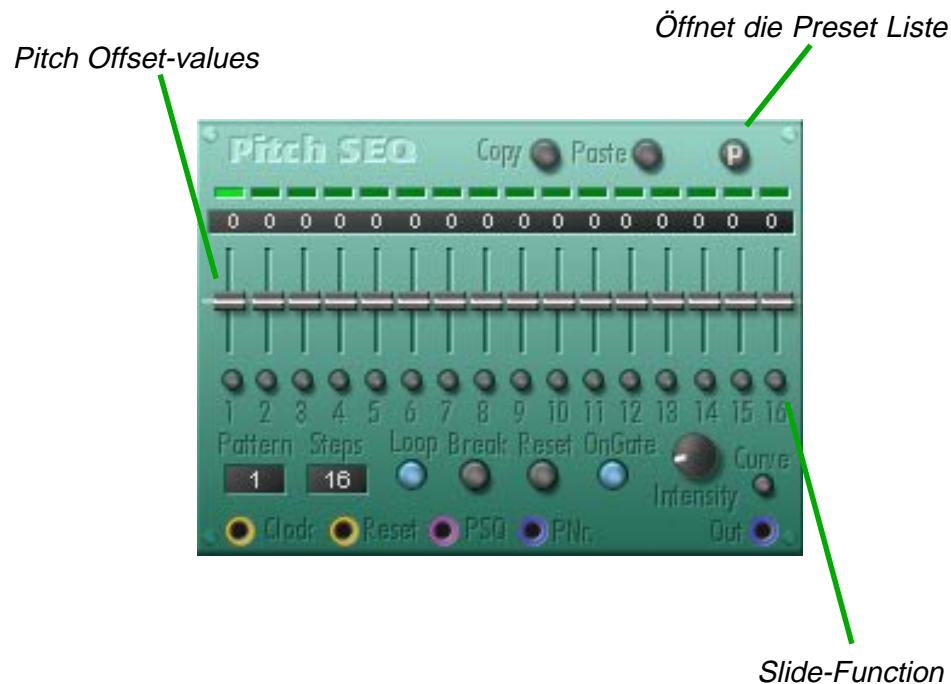
Connect the PS32 Pattern Switcher here to permit selection of patterns from the keyboard.

### Out

The unipolar modulation signal is output here.

## Pitch SEQ

The Pitch Sequencer is largely similar to the Ctrl SEQ B2 bipolar control sequencer, but is optimized for pitch modulation. It produces a bipolar control signal for connection to the pitch modulation input of a pitch modifier module, with modulation values in the range -36 – +36 semitones. At your option, the pitch value programmed for each step can either always be sent or sent only when a gate occurs on that step. Finally, the slide buttons permit step-by-step control over the change from one pitch value to the next, which can be either an instantaneous change or a smooth glide.



## Copy

Click here to copy the current sequence into the clipboard.

## Paste

Click here to replace the current sequence with a sequence previously copied into the clipboard. This exchange is possible only within a single sequencer module.

You can also paste a copied sequence after switching to a new preset, which permits transfer of sequences from one preset to another.

## Preset

Click the P button to open the module's preset list. Each preset contains 32 independent sequences.

## Pitch Offset Values

Use these faders to set control values for each step in a pattern (or enter values in the range -36 to +36 directly via the text field above each fader). Pitch Offset-Wert

## Slide

These three-state buttons permit you to activate a control-value glide function on a step-by-step basis. The glide can be either exponential (blue) or logarithmic (yellow). Which type is to be preferred depends upon the parameter being modulated – for example, filters often sound better with exponential control.

Depending upon the tempo and your own personal taste, you may also need to adjust the Intensity control (see below) to get the glide function to sound the way you want.

## Intensity (Ctrl SEQ B2 only)

Controls the "steepness" or speed of the exponential or logarithmic control-value glide in pattern steps for which the Slide function is active (all steps are affected in common).

## Curve (Ctrl SEQ B2 only)

Same as the Slide buttons described above – but switches the states of these buttons in common for all sixteen pattern steps at one time.

## Pattern

Here, you specify which of the 32 available patterns is to be played. (see also Gate SEQ)

## Steps

Pattern length is independently variable for each pattern, up to a maximum of 16 steps. (see also Gate SEQ)

## Swing

Controls the amount of rhythmic "swing". (see also Gate SEQ)

## Loop

Activate this function to cause the pattern to repeat continuously. (see also Gate SEQ)

## Break

Clicking on this button stops the sequencer.

## Reset

Click on **Reset** to restart the pattern from step 1. This can also be done from outside via the Reset input – permitting restart to be triggered by notes played on a keyboard, for example. (see also Gate SEQ)

## OnGate

Activate this option to cause control values to be sent only on steps in which a gate event (note-on) occurs. For this to work, the Ctrl SEQ Clock must come from the Gate SEQ module and not directly from the MIDI Clock module or Clock Divider module.

When this option is *not* active, control values are sent on every step, regardless of whether a note-on occurs, meaning that new control values may change the sound of an already-playing note from an earlier step.

## Connections

### Clock

Connect the clock output of the MIDI Clock module here – or better still, the output of a Clock Divider.

### Reset

Connect a trigger signal here to permit remote-actuated restarting of a pattern from step 1.

### PSQ

For connection to the signal output of the same name on the Pattern Sequencer, which can then remote-control pattern number and pattern length.

### PNo.

Connect the PS32 Pattern Switcher here to permit selection of patterns from the keyboard.

### Out

The bipolar modulation signal is output here.

## Pattern SEQ

The pattern sequencer permits remote-control selection of the pattern to be played by a step sequencer. Connection of the pattern sequencer overrides the local pattern-select and number-of-steps controls, so that the pattern sequencer also directly controls pattern length, regardless of the pre-programmed length of the pattern being played. The pattern sequencer permits pre-programmed looping of a handful of patterns or the creation of complete song structures.

To modify an entry in the Pattern List, simply select it and enter the new value directly from your computer keyboard. A selected entry in the list allows itself to be overwritten directly – just start typing the new value.



As soon as the Pattern Sequencer sends a pattern change command, each connected sequencer pre-loads its next pattern, switching to the new pattern when the current pattern has played through to the end. Note that pre-loading does require a small amount of time. In order to ensure that the transition to the new pattern is made correctly, you should avoid setting playback pattern length too short. As a rule, this is a problem only when the pattern length is set below four steps, and when the playback tempo is high.

### Pattern List

The Pattern List lets you control the sequence of patterns to be played and the actual playback length of each pattern.

Each entry in the Pattern List shows the song step number, the name and number of the selected pattern and the playback length of the pattern in steps.

When song mode is not active, the connected step sequencers can also be made to switch to new patterns by clicking on the desired list entry. As always, the actual switch to the new sequence takes place when the current pattern has played through to the end.



## Song Mode

Activate this option to cause the pattern sequencer to work its way through the pattern list. When deactivated, only the currently selected pattern plays.

**In song mode, switching to new sequences by clicking on entries in the Pattern List is not possible. In this case, the Pattern SEQ module exclusively controls pattern transitions. To jump to a different position, first deactivate song mode, select the desired song step and then reactivate song mode.**

## Song Step

The pattern sequencer can handle songs consisting of up to 256 patterns. This field shows which pattern of a song is currently playing.

## Start

Click this button to start the pattern sequencer.

**This applies also when you're not playing a song – it is nonetheless necessary, so that the pattern sequencer will send clocks to the connected sequencer modules, which otherwise won't run.**

## Stop

Click this button to stop the pattern sequencer, which ceases sending clocks. If you now click the Start button, song playback will resume where it left off. Clicking Stop a second time will reset the song to the beginning.

## Loop

Activating this option causes the current song or pattern-chain to repeat indefinitely.

## Loop Length

Sets the number of song steps (starting with step 1) which are to be included in a song loop. For example, if this is set to 4, then the first four patterns of the song will repeat indefinitely.

## Pattern

Specifies the pattern number which is to be sent to the connected step sequencer for the currently-selected song step.

**Alternatively, pattern numbers can be entered directly into the pattern list. To do this, select the existing pattern number at the desired song step and enter a new pattern number from the keyboard.**

## Steps

Specifies the number of repetitions of the pattern at the current song step. As with pattern numbers, these values can also be entered into the pattern list directly (when doing so, it's a good idea to stop the pattern sequencer first!)



## Connections

### Clock In

Connect the clock output of the MIDI Clock module here – or better still, the output of a Clock Divider.

### Sync

Triggers advance to the next step in the pattern list. A typical connection would be to the Sync output of the MDS8 sequencer, which sends a sync signal upon reaching the end of its current pattern.

### Reset

Connect this signal to other step sequencers in order to have them simultaneously reset when the pattern sequencer sends a stop signal.

### Link

Connect the Link output of the pattern sequencer to the Link input of the MDS8 in order to transmit the clock signal. The Start/Stop status of the pattern sequencer is likewise taken into account.

### PSQ

For connection to the input of the same name on a step sequencer, so that its pattern length and number of steps can be remote-controlled by the pattern sequencer.

## PS32

This module permits you to call up new patterns without having to do so directly on the sequencer module itself. More interestingly, however, it is also possible to remote-control bank and pattern select on the PS32 from a MIDI keyboard. In combination with a MIDI KeySplitter module, you can play normally in one part of the keyboard while selecting patterns in another part.



### Bank

Select the desired bank here.

#### 1-4/5-8

Here, select one of the eight patterns in a bank.

### Set

Opens the Settings dialog, in which you can assign MIDI note numbers to buttons for remote-control purposes.

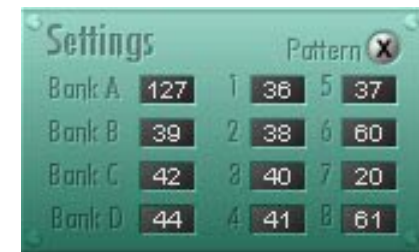
## Connections

### MIDI

MIDI input – connect it directly to a MIDI source, or to one of the outputs of a KeySplitter module.

### Out

Connect this signal to the PNr input of one or more sequencer modules.



### Bank A-C

Specify the desired MIDI note numbers here.

### Pattern 1-8

Specify the desired MIDI note numbers here.

## GateOr

For use in conjunction with the MVC module. A "gate collector" which permits derivation of a trigger signal every time a new note is played on the keyboard – for example, to send a Reset to a step sequencer module whenever a new note is played, causing it to restart its current pattern.



### Gate In

Connect the Gate output of the MVC module here.

### Gate Out

Connect this signal to the Reset input of a step sequencer, for example, so that it restarts its pattern whenever a new note is played on the keyboard.

## Start/Stop

This is a simple module for starting and stopping the sequencer when, for example, no pattern sequencer is patched into the circuit. Connect the Out of this module to the start/stop input of a Clock Divider.



### Controls

#### Start/Stop

When this module is connected to a clock divider, this control starts and stops any sequencer modules also connected to the clock divider.

### Connections

#### Out

Start/Stop signal output.

# Index

## A

Active 129  
Adjusting Levels 39  
Adjusting Times 39  
ADSR 30  
Aftertouch 3  
Amod 32, 33, 35, 37  
Amplitude Modulator 50  
ASlope 32, 34, 36, 110  
Att 45, 46  
Attack 41, 116, 117  
Audio-Trig 85

## B

Bedienelemente 39  
Bit 127  
Bit on/off 127  
Bit Quantizer 127  
BPF 25  
Bypass 119, 121, 122, 124

## C

CFm1  
57, 60, 61, 62, 64, 66, 67, 69  
CFm2  
57, 58, 59, 60, 61, 62, 63, 67, 69  
Changing the Mode 39  
Chorus 121  
Clock 8, 93  
Coarse 13  
Coarse/Fine  
2, 15, 17, 19, 20, 21, 22, 24, 26, 87  
Combfiler A/B 68  
Compressor 116  
Constant Value 90

Constant Value bipolar 90  
Controls 39  
Crossfade 47, 49  
Crossfade Modulator 48  
Curve 91  
Curve Table 91  
Cutoff

55, 57, 58, 59, 60, 61, 62, 63, 64, 66, 67, 68

## D

Damp 24, 68  
DampMod 24, 68  
db/Oct 64  
Decay 103  
Decimator 127  
Defining a Loop 39  
Delay 78, 79, 118  
Delete button 39  
Deleting Segments 39  
Depth 121, 122, 124  
Detune 23  
Dist 126  
Distortion 126  
Dmod 24, 33, 37  
Drum 97  
Drum Oscillator 101  
Drum Synth 97  
Drum Voice Control 105  
Dry 129  
Dry/Wet 122, 124

## E

EG 23  
eight-note triplets 8  
Envelope Follower 41  
Envelopes 29

EQ 72  
Esync 4, 31, 33, 35, 37, 40, 106  
ExpM 89  
Exponential VCA 43  
Ext 76, 78, 80, 84

## F

Fade In 76, 79  
Fade Out 78  
FB 118, 119, 122, 124  
Filter 55  
Fine 13  
fingered Glissando (fG) 3  
fingered Portamento (fP) 3  
Fixed 22  
Flanger 122  
Fm 1 23  
Fm Operator 22  
FmA1/2 22, 27  
Freq 72, 73, 74, 77, 78, 81, 90  
Freq Divide 93  
Freq In  
13, 15, 18, 19, 21, 23, 24, 87, 88, 89, 94  
Freq Out 87, 88, 89, 94  
Frequency & Pitchmodulation 12  
Frequency Divider 93

## G

Gain 72, 73, 74, 116, 117  
Gain 1 - 4 52  
Gate  
31, 33, 35, 37, 77, 78, 81, 85, 100, 102, 104, 105  
Gate In 23  
Gate Out 106  
Gate/Trigger 85  
Glissando (G) 3

## H

Hard 126  
HiDamp 118, 119  
High Key 6  
Hihat Source 108  
Hmod 35  
HPF 67, 108

## I

In 24, 85  
In 1- 4 53  
In L/m 120, 121, 122, 124  
In R 120, 121, 122, 124  
Init Phase 76, 78, 79, 80  
InL/m 129  
Input Gain 41  
InR 129  
Insert-Slot 129  
Inserting Segments 39  
Internal/External 7

## K

Key Zone 6  
Keyf  
55, 59, 60, 61, 62, 63, 64, 66, 67, 69, 77, 81, 87

## L

Level 84  
LFO 79, 80, 82  
Limiter 117  
Linear VCA 43  
LinM 89  
Lmod 30, 34, 36  
Loops 39  
Low Key 5, 6  
Lowpass 57

## M

Man/Mod 17  
MIDI 2  
MIDI High 5  
MIDI In  
    4, 5, 9, 100, 102, 104, 105  
MIDI Low 5  
MIDI Out 6  
MIDI to Trigger 9  
Mix & Gain 42  
Mix 2/4/8 47  
Mod 48, 49  
Mod1 50  
Mod2 50  
Mode 64  
Modifier 86  
Mono/Stereo Insert 129  
Morphing Pulse 19  
Morphing Saw 20  
Multi LFO A 76  
Multi LFO B 78  
Multi OSC 13  
Multimode Filter A 64  
Multimode Filter B 66  
Multisegment Envelopes 38  
Mute In 100, 105  
MVC 2  
MW LFO 79

## N

Noise 108  
Noise Attack 99  
Noise Cf 99  
Noise Cf Mod 99  
Noise Color 102  
Noise Decay 99, 102  
Noise Generator 25

Noise Level 99, 108  
Noise Res 99  
Noise Slope 102  
NoiseL 102  
Note No. 101

## O

Offset 91  
On/Off Switch 52  
OSC 12  
Osc 108  
Out 13, 15  
Out 1- 4 52  
Out L 48, 49  
Out R 48, 49  
Output Gain 41

## P

P 17  
Pan L/R 48, 49  
Pan Modulator 49  
Parametric EQ 72  
PDec 103  
Peak Meter 54  
Peak Textfeld 54  
Phaser 124  
Pitch Modifier B 88  
Pitch Modifier C 89  
Pitch Quantizer 95  
Pitch Wheel Range 2  
PMod 102  
PMod1 87, 88  
PMod2 87, 88  
Portamento (P) 3  
PP 17  
PPm 18  
Pulse LFO 80  
Pulse OSC 14

PW 13, 17, 19  
PWidth 80  
Pwm 13, 18, 19  
PwmA 17, 19, 80  
PWR 2

## Q

Quant 95

## R

Random Signal Generator 84  
Range 95  
Rate  
    76, 78, 79, 80, 82, 83, 84, 121, 122, 124  
Ratio 116, 117  
Release 116, 117  
ResM 64, 67  
ResMod 24, 68  
Resonanz 64, 66, 67, 68  
Ret 27  
Retrig 76, 79, 80  
Ringmodulator 128  
Rmod 24, 37  
Rmod1 76, 80  
Rmod2 76, 77

## S

S 17  
Sample & Hold 85  
Sample Rate 127  
Saw Down LFO 82  
Saw down/up 15  
Saw Up LFO 82  
Saw/Pulse 21  
Schwellwert 85  
Selecting a Node 39  
Sensitivity 91

Set button 39  
Sine 1 809 98  
Sine 1 Level 99  
Sine 1 PDec 98  
Sine 1 PMod 98  
Sine 1 Snap 98  
Sine 1Tune 98  
Sine 2 Decay 98  
Sine 2 Detune 98  
Sine 2 Level 99  
Sine 2 Slope 99  
Sine 809 101  
Sine Decay 101  
Sine Level 101  
Sine OSC 16  
Sinus LFO 82  
Slope 30, 40, 43, 44  
Smod 37  
Snap 104  
Soft 126  
SP 17  
Spectral OSC 21  
SPm 18  
Square LFO 83  
Start 147  
Start/Stop 8, 147  
Static Crossfade 47  
Static Pan 48  
Stereo Insert 129  
Stop 147  
Sub 17  
Switch 52  
Switch 1 - 4 52, 53  
Sync In 13, 15  
Sync Out 18  
SyncM 13

SyncMaster 12  
SyncS 13, 14  
SyncSlave 12  
Synthese-Effekt 126

## **T**

Threshold 85  
Time 118, 119  
Tmod1 30, 32, 33, 34, 35, 36, 37, 40  
Tmod2 30, 32, 33, 34, 35  
Transpose 6  
Triangle LFO 83  
Tube Resonator 24  
Tune 102  
Tune 1-3 108

## **U**

Uknow Filter 67  
Uknow OSC 17

## **V**

Val 90, 91  
Vel 98, 101, 103, 105  
Velocity/Aftertouch 3  
Volume Attenuator 45

## **W**

Waveform 13, 76, 78  
Wet 129  
WF 19, 20  
Wfm 19, 21  
WfmA 20, 21  
White 25

## **X**

X-Fade 1/2 47, 48  
Xmod & Feedback Connector 96