

# ModulAir

## Polyphonic Modular Synthesizer

Version 1.2

© 2018-2020 by Björn Arlt @ Full Bucket Music  
<http://www.fullbucket.de/music>



VST is a trademark of Steinberg Media Technologies GmbH  
Windows is a registered trademark of Microsoft Corporation  
The Audio Units logo is a trademark of Apple Computer, Inc.

## Table of Contents

Introduction.....	4
Acknowledgments.....	4
Architecture.....	5
ModulAirFX (VST).....	5
Common Section.....	6
Modules.....	7
Module Patching And Source Highlighting.....	7
Oscillators Modules.....	8
Oscillator.....	8
Dual Oscillator.....	8
Linear FM Oscillator.....	9
Formant Oscillator.....	9
Wavetable Oscillator.....	10
Dual Noise.....	10
Sample Player.....	11
Filter Modules.....	12
Filter (Ladder).....	12
Filter (K35).....	12
Filter (Comb).....	13
Resonators.....	13
St.Filter (Ldr.), St.Filter (K35), St.Filter (Comb), St.Resonators.....	14
Equalizer.....	14
Parametric EQ.....	14
Filter Bank.....	15
Amplifier Modules.....	16
Amplifier.....	16
Mixer.....	16
Stereo Mixer.....	17
Signal Processor.....	17
Blender.....	18
Processor Modules.....	19
Ring Modulator.....	19
Audio S/H.....	19
Hilbert.....	20
Bit Reducer.....	21
Dual Saturator.....	21
Analyzer.....	22
Modulator Modules.....	23
Envelope.....	23
Dual Envelope.....	23
Quad Envelope.....	24
Dual Linear Envelope.....	24
LFO.....	25
Dual LFO.....	25
Sample & Hold.....	26
Sequencer Modules.....	27
Sequencer.....	27
Beat Sequencer.....	27
Sequential Switch.....	28
Audio Switch.....	28

---

Sequencer Clock.....	29
Effect Modules.....	30
Phaser.....	30
BBD Delay / Stereo BBD Delay.....	30
Dual Delay Line.....	31
Reverb.....	31
Tool Modules.....	32
Voltage Processor.....	32
LAG/Adder.....	32
Sample Delay.....	33
Modulation Signal Delay/M2G.....	33
Gate Tools.....	34
Oscilloscope.....	34
Keyed Gate, Keyed Modulation, Keyed Signal.....	35
Miscellaneous Functionality.....	36
Options Menu.....	36
MIDI Learn.....	36
Custom Tuning.....	36
MIDI CC Sources.....	37
Frequently Asked Questions.....	38

## Introduction

*ModulAir* is a polyphonic modular software synthesizer plug-in for Microsoft Windows (VST) and Apple macOS (VST/AU). It is written in native C++ code for high performance even on "lighter" systems. The main features are:

- Fully modular architecture
- Up to 18 modules per patch
- Polyphonic (up to 64 voices) and monophonic *Master* modules
- External signal processing possible
- TUN/SCL/KBM micro-tuning file import
- MIDI Learn – all parameters can be controlled by MIDI CC
- Double precision audio processing
- Plug-in supports Windows and macOS (32 bit and 64 bit)

*ModulAir* is (although officially released) a **work in progress** with only a bunch of quirky presets and a lousy documentation. Note that I am constantly adding modules and stuff to it. ☺

If you experience any problems that you cannot resolve by following the suggestions in the →*Frequently Asked Questions*: Please send a mail to `full.bucket@gmx.net`.

**Please tell me what OS (Windows, MacOSX, 32 or 64 bit) and Host/DAW software you are using!** Also send me a mail if you have suggestions or comments.

**And please note:** I will not add "virtual cables" dangling on the screen. Instead I will stay with the "plug" concept that I already use with my other plug-ins (*FB-3X00* series, *blooo*, *scrooo*, *qyooo*, etc.).

## Acknowledgments

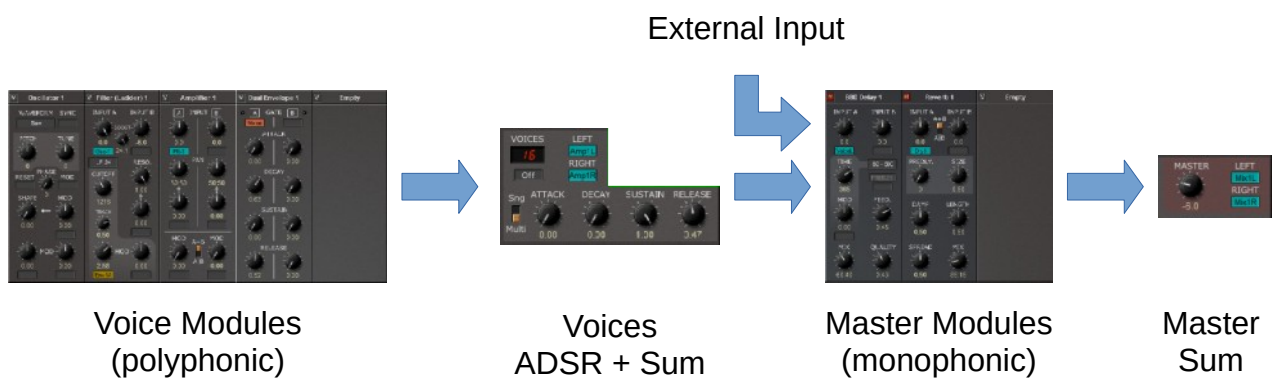
- First I wanted to thank Cockos and **Oli Larkin** for developing and maintaining the WDL(-OL)/IPlug framework, and **Laurent Bergman** for his French manuals.
- The TUN/SCL file handling source code is adapted from **Mark Henning** at <https://www.mark-henning.de/>.
- Furthermore a **BIG THANKS** to **Tim Stinchcombe** and **Will Pirkle** for their in-depth analysis of the K35 filter chip.
- Last not least another THANK YOU! to kraftraum, Manfred Hasenfus and to the KVR Audio community, especially to `bjporter`, `BlackWinny`, `Halonmusic`, `Dee.P.Tree`, `fmr`, and `martin_l`.

## Architecture

*ModulAir* is a modular synthesizer that can hold up to 18 modules. Each module can either be a polyphonic *Voice module* or a monophonic *Master module*:

- *Voice modules* are used for the individual synthesizer voices (typically oscillators, filters, modulators etc.).
- *Master modules* are used for overall sound processing of all voices (e.g. effects like delay or reverb).

However, there is no limitation in the way that modules are used – it is perfectly OK to create a (in this case necessarily monophonic) synthesizer completely out of Master modules. And it is also possible to add *Reverb* as a Voice module (which then of course will only reverberate the individual voice). The following diagram shows the signal flow of *ModulAir*.



The voice signals generated by the Voice modules are shaped by individual ADSR envelope generators and then summed up to form the *Voices Sum* stereo signal. This signal along with an external input signal can be used to feed the Master modules. The final output can then be selected from the output of a Master module (if applicable) or the *Voices Sum* signal.

While audio signals cannot be sent directly from Voice modules to Master modules or vice versa, it is possible to send modulation and/or gate/trigger signals from Master modules to Voice modules. This way one can create global LFOs or sequencers and trigger sources.

## ModulAirFX (VST)

In some VST hosts you cannot use *ModulAir* as an VST effect (due to the definition of the VST 2.4 protocol). The *ModulAirFX* VST component is a simple wrapper for *ModulAir* to circumvent this problem – just make sure to install the *ModulAirFX* component into the same folder as *ModulAir* itself.

## Common Section

The Common section is used to select and (re)name the current program, apply →MIDI Learn, and invoke the →Options Menu. Furthermore you can set the following parameters: *Pitch Bend* range ( $\pm 24$  notes), *Portamento* time and mode (always on or *Legato* only), →*Custom Tuning*, →*MIDI CC Sources*, and the number of voices (polyphony). Using the *Drone* button right below the *Voices* control you can “play” a droning note without sending any MIDI input to *ModulAir*.

To send a signal to the left and/or right *Voices Sum* signal it has to be “plugged” into the *LEFT* and/or *RIGHT* plugs of the gray *Voices* section. Clicking on a plug opens a menu where you can select the signal source that you want to assign to the plug. The *LEFT/RIGHT* plugs will glow red if no signal is connected to them.



The volume envelope of a voice signal is shaped by an ADSR generator with adjustable trigger mode (*Single* or *Multiple*). Thus, you do not have to add extra Amplifier and Envelope modules (although you can) for your sound patch. If you turn *Release* to the very right (i.e. to 1.00), the ADSR goes into *HOLD* mode. Note that this may have a heavy impact on CPU performance!

Whatever signal should be sent to the final output of *ModulAir* has to be plugged into the *LEFT/RIGHT* plugs of the light-red Master section. Again, these plugs will glow red if no signal is connected to them – and in this case you will not hear any sound on the respective stereo channel! The final output volume is controlled by the *Volume* knob.

## Modules

Currently there are 55 different module types available of which 18 different modules can be used per program at the same time. The module types are divided into the following groups:

- Oscillators (7 types)
- Filters (7 + 4 stereo versions)
- Amplifiers (4 + 1 stereo version)
- Processors (6)
- Modulators (7)
- Sequencers (5)
- Effects (4 + 1 stereo version)
- Tools (9)

You can add/change a module by clicking the header of the respective slot and selecting the desired module from a context menu. If you right-click the header you can move the module to a different slot. Alternatively you can drag & drop the module to the new location. If you drag it to the Common section, its outputs will be automatically plugged to the output plugs of the respective section (*Voice* or *Master*). Of course this will only work for modules that feature at least one audio output (like oscillators, amplifiers or filters).

Clicking on the top-left button (labeled with "V" for *Voice* or "M" for *Master*) will change a *Voice* module to a *Master* module or vice versa.

## Module Patching And Source Highlighting

To patch a module plug just click on the plug symbol and select the required source signal from the context menu. Audio signals are color-coded in blue, modulation signals in yellow, and Gate signals in orange.

When you hover over a module plug, the source module of the plug (or its subsection when the module offers more than one output) will be highlighted in the respective color code. You can turn this feature of in the *Options Menu*.





## Oscillators Modules

### Oscillator

The *Oscillator* module is a standard band-limited oscillator with the following parameters:

Waveform: Sets the waveform (*Sine*, *Saw*, *Pulse*, *Triangle*, *Noise*).

Sync: Source oscillator for Hard Sync.

Pitch: Shifts the frequency up or down in semitones by a maximum of two octaves.

Tune: Fine tuning in cents of plus/minus one semitone.

Reset: Gate/trigger source for resetting the oscillator's phase.

Phase: Phase (in degrees) to which the oscillator will be reset.

(Phase) Mod: Source for reset phase modulation.

Shape: Shape of the waveform. If *Pulse* is selected this will correspond to pulse width modulation.

(Shape) Mod: Source and amount for shape modulation.

(Frequency) Mod: Source and amount for frequency modulation.



The *Oscillator* module has one audio and one *Sync* output.

### Dual Oscillator

The *Dual Oscillator* module features two band-limited oscillators *A* and *B* with the following parameters:

Waveform A/B: Sets the waveform (*Sine*, *Saw*, *Pulse*, *Triangle*, *Noise*).

Pitch A/B: Shifts the frequency up or down in semitones by a maximum of two octaves.

Tune A/B: Fine tuning in cents of plus/minus one semitone.

Shape A/B: Shape of the waveform. If *Pulse* is selected this will correspond to pulse width modulation.

Mod A/B: Source and amount for frequency modulation.

The *Dual Oscillator* module has two audio and two *Sync* outputs (one per oscillator *A* and *B*).





## Linear FM Oscillator

The *Linear FM Oscillator* module is a band-limited oscillator that allows for linear frequency modulation. It has the following parameters:

Waveform: Selects the waveform (*Sine*, *Saw*, *Pulse* or *Triangle*).

Sync: Source oscillator for Hard Sync.

Pitch: Shifts the frequency up or down in semitones by a maximum of two octaves.

Tune: Fine tuning in cents of plus/minus one semitone.

Reset: Gate/trigger source for resetting the oscillator's phase.

Phase: Phase (in degrees) to which the oscillator will be reset.

(Phase) Mod: Source for reset phase modulation.

Lin. FM: Source and amount for linear FM.

(Lin. FM) Mod: Source and amount for linear FM modulation.

(Frequency) Mod: Source and amount for frequency modulation.



The *Linear FM Oscillator* module has one audio and one *Sync* output.

## Formant Oscillator

The *Formant Oscillator* module is an oscillator that generates waveforms with an adjustable spectrum relative to the base frequency or a fixed frequency. It has the following parameters:

Mode: Selects the oscillator mode.

- *Fixed*: The spectrum will be centered at a fixed frequency.
- *Partial*: The spectrum will be centered relative to a partial of the base frequency.
- *Ring*: The spectrum will be centered relative to a fractional multiple of the base frequency.

(Formant Frequency): Fixed frequency/partial/multiple of the center of the spectrum.

Spread: Adjusts the bandwidth of the generated spectrum.

(Formant Frequency) Mod: Source and amount for Formant frequency modulation.

Pitch: Shifts the frequency up or down in semitones by a maximum of two octaves.

Tune: Fine tuning in cents of plus/minus one semitone.

(Frequency) Mod: Source and amount for frequency modulation.



The *Formant Oscillator* module has one audio and one *Sync* output.

## Wavetable Oscillator

This module is a band-limited wavetable oscillator with the following parameters:

**Waveform:** Selects the current wavetable. The *Wavetable Oscillator* can read *Serum WAV* and *Waldorf Blofeld SysEx* wavetable files. Any other WAV file will be treated as a wavetable of 2048 samples with at most 64 subsequent wavetable entries.

**Sync:** Source oscillator for Hard Sync.

**Pitch:** Shifts the frequency up or down in semitones by a maximum of two octaves.

**Tune:** Fine tuning in cents of plus/minus one semitone.

**Reset:** Gate/trigger source for resetting the oscillator's phase.

**Phase:** Phase (in degrees) to which the oscillator will be reset.

**(Phase) Mod:** Source for reset phase modulation.

**Shape:** Shape of the waveform. If *Pulse* is selected this will correspond to pulse width modulation.

**(Shape) Mod:** Source and amount for shape modulation.

**(Scan Mode):** Sets the mode of transition when scanning the wavetable. *Interpolation* will interpolate between the different waves while *Raw* will perform hard-switching.

**(Frequency) Mod:** Source and amount for frequency modulation.



The *Oscillator* module has one audio and one *Sync* output.

## Dual Noise

The *Dual Noise* module features two noise generators *A* and *B* with the following parameters:

**White A/B:** Level of the white noise.

**Pink A/B:** Level of the pink noise.

**(White Mod A/B):** Source and amount for white noise amplitude modulation.

**(Pink Mod A/B):** Source and amount for pink noise amplitude modulation.

The *Dual Oscillator* module has five audio outputs (*White Noise A* and *B*, *Pink Noise A* and *B*, and the sum of all four noise signals).



## Sample Player

The *Sample Player* module allows the play back of an audio file in WAV format (mono/stereo).

**Sample:** Selects the current audio/sample WAV file.

**Mode:** Sets the sample reconstruction mode (*Resample* or *Sample Drop*). *Resample* mode is the (recommended) default and causes less aliasing while *Sample Drop* is the naive "lo-fi" variant as used in some old samplers.

**Tune:** Fine tuning in cents of plus/minus one semitone.

**Keyed/Fixed:** Sets if the sample playback pitch should follow the current note played or if it should be fixed. In *Keyed* mode you can select the reference key at which the sample will be played at its original pitch. In *Fixed* mode you can adjust the playback pitch between -48 and +48 notes.

**Trigger:** Selects the source that (re-)triggers the sample playback.

**Start & End:** Sets the start and end points of the sample. If *Start* is higher than *End* the sample is played backwards.

**Start Mod:** Selects the source for the modulation of the sample start point.

**Start Mod Amount:** Sets the amount of the modulation of the sample start point.

**Loop:** Selects the sample loop mode (Off, On or Alternating).

**Loop Start & End:** Sets the start and end points of the loop. If *Loop Start* is higher than *Loop End* the loop is played backwards.

**(Frequency) Mod:** Source and amount for frequency modulation.

Right-clicking the *Start*, *End*, *Loop Start* or *Loop End* knobs opens a small window where you can adjust the respective parameters (sample-exact).

The *Sample Player* module has two audio outputs (*Left* and *Right*).



## Filter Modules

### Filter (Ladder)

The *Filter (Ladder)* module is a self-resonating multi-mode ladder filter with the following parameters:

Input A/B: Level and source of the input signal.

Boost: Boosts the input signal at the input stage of the filter and attenuates the output by the same amount.

Mode: Filter mode (*Bypass*, *Lowpass* 6–12–18–24 dB/Oct., *Highpass* 6–12–18–24 dB/Oct., *Bandpass* 6–12 dB/Oct., *Allpass* 6–12–18–24 dB/Oct.).

Cutoff: Cutoff frequency of the filter.

Resonance: Resonance/feedback level.

(Resonance Mod): Source and amount for resonance modulation.

Track: Amount of keyboard tracking.

(Frequency) Mod: Source and amount for cutoff frequency modulation.



The *Filter (Ladder)* module has one audio output.

### Filter (K35)

The *Filter (K35)* module is a multi-mode filter modeled after the legendary K35 filter chip with the following parameters:

Input A/B: Level and source of the input signal.

Boost: Boosts the input signal at the input stage of the filter and attenuates the output by the same amount.

Mode: Filter mode (*Bypass*, *Lowpass*, *Highpass*).

Cutoff: Cutoff frequency of the filter.

Resonance: Resonance/peak level.

(Resonance Mod): Source and amount for resonance modulation.

Track: Amount of keyboard tracking.

(Frequency) Mod: Source and amount for cutoff frequency modulation.



The *Filter (K35)* module has one audio output.

## Filter (Comb)

The *Filter (Comb)* module is a comb filter (a very short delay with variable delay time) with the following parameters:

Input A/B: Level and source of the input signal.

Boost: Boosts the input signal at the input stage of the filter and attenuates the output by the same amount.

Mode: Feedback mode (*Positive, Negative*).

Cutoff: Cutoff frequency of the filter.

Resonance: Resonance/feedback level.

(Resonance Mod): Source and amount for feedback modulation.

Track: Amount of keyboard tracking.

CLR: Gate/trigger source for clearing the delay filter.

(Frequency) Mod: Source and amount for cutoff frequency modulation.



The *Filter (Comb)* module has one audio output.

## Resonators

The *Resonators* module features three resonating filters known from the Korg PS-3100/3300 synthesizers with the following parameters:

Input A/B: Level and source of the input signal.

Mix: Mix between input and filtered signal.

(Frequency) A: Center frequency of resonator A.

(Frequency) B: Center frequency of resonator B.

(Frequency) C: Center frequency of resonator C.

(Frequency) Mod: Source and amount for center frequency modulation.

The *Resonators* module has four audio outputs (*Mix, Resonator A, Resonator B, Resonator C*).





### St.Filter (Ldr.), St.Filter (K35), St.Filter (Comb), St.Resonators

These four modules are stereo versions of the filter modules mentioned above.

Left/Right: Level and source of the left/right input signal.

For additional parameters see the respective "mono" modules.

The stereo modules have exactly **two** audio outputs (*Left* and *Right*).

### Equalizer

The *Equalizer* module is a 10-stage graphic equalizer with the following parameters:

Input A/B: Level and source of the input signal.

(Mono/Stereo): Switch from mono (A+B) to stereo input (A|B).

Frequency Bands (16kHz – 30Hz): Level of the respective frequency band in decibel.

The *Equalizer* module has two audio outputs.



### Parametric EQ

The *Parametric EQ* module is a 5-stage parametric equalizer with the following parameters:

Input A/B: Level and source of the input signal.

(Mono/Stereo): Switch from mono (A+B) to stereo input (A|B).

Frequency 1 – 5 (16kHz – 16000Hz): Frequency of the respective band in Hertz.

Level 1 – 5: Level of the respective frequency band in decibel (-18 to +18 dB).

The *Parametric EQ* module has two audio outputs.



## Filter Bank

The *Filter Bank* module is a 10-stage resonator bank typically used as the synthesis block of a vocoder in conjunction with the *Analyzer*. It has the following parameters:

***Input***: Level of the left input signal.

***Left(M)***: Source of the left input signal. If no source is plugged into the *Right* plug, the *Filter Bank* works in Mono mode.

***Right***: Source of the right input signal.

***Shift***: Shifts all frequency bands up or down.

***(Sharpness)***: Resolution of the resonators (*SOFT* or *SHARP*).

***Frequency Bands (A – J)***: Frequency of the respective frequency band in Hertz along with its gain modulation source.

The *Filter Bank* module has two audio outputs.





## Amplifier Modules

### Amplifier

The *Amplifier* module is a dual/stereo amplifier with the following parameters:

Input A/B: Level and source of the input signal.

Pan A/B: Panorama of the input signal.

(Pan Mod) A/B: Source and amount for panorama modulation.

Mod A/B: Source and amount for amplitude modulation.

(Mode):

- $A+B$  means that the two modulation sources affect both input signals  $A$  and  $B$ .
- $A|B$  means that the left modulation source only affects input  $A$  while the right modulation source only affects input  $B$ .



The *Amplifier* module has two audio outputs (*Left* and *Right*).

### Mixer

The *Mixer* module is a four-channel mixer with the following parameters:

Input: Level and source of the respective input signal.

Pan: Panorama of the respective input signal.

Output: Total output volume.

The *Mixer* module has two audio outputs (*Left* and *Right*).



### Stereo Mixer

The *Stereo Mixer* module is a four-channel stereo mixer with the following parameters:

Input L+R: Source and level of the respective input signal.

Output L+R: Total output volume of the respective channel.

The *Stereo Mixer* module has two audio outputs (*Left* and *Right*).



### Signal Processor

The *Signal Processor* module features four amplifiers with the following parameters:

Input A/B/C/D: Source of the input signal.

Mod A/B/C/D: Source and amount for amplitude modulation.

The *Signal Processor* module has five audio outputs (*Signal A* to *D* and *Summed Output*).



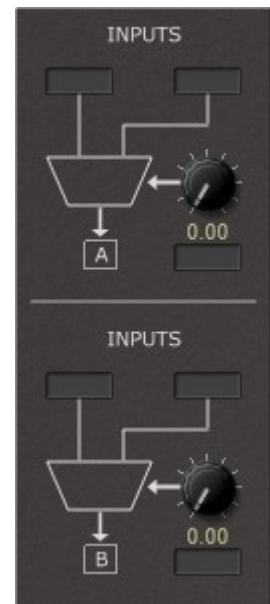
## Blender

The *Blender* module features two amplifiers that can blend between two signal inputs. It has the following parameters:

*Inputs A/B*: Source of the two input signals of each section.

*(Blend) A/B*: Source and amount for blending between the two inputs of each blender section.

The *Blender* module has two audio outputs (*Blender A* and *B*).



## Processor Modules

### Ring Modulator

The *Ring Modulator* module features two Ring modulators with the following parameters:

Input A1/A2: Sources of the inputs for Ring modulator A.

Input B1/B2: Sources of the inputs for Ring modulator B.

Amount: Intensity of Ring modulation.

Mod: Source and amount for Ring intensity modulation.

The *Ring Modulator* module has two audio outputs (*Ring Modulator A* and *B*).



### Audio S/H

The *Audio S/H* module resamples the incoming signal at a different rate. It has the following parameters:

Input: Source of the input.

Tonality: Typically, the resampled signal is constructed of band-limited pulse steps. With this control the steps can be gradually changed to a saw-like shape.

(Fixed Frequency On/Off): Switches between keyboard-controlled and fixed frequency.

(Fixed Frequency): Fixed resample frequency (when fixed frequency is on).

Pitch: Shifts the resample frequency up or down in semitones by a maximum of two octaves (when fixed frequency is off).

Tune: Fine tuning of the resample frequency in cents of plus/minus one semitone (when fixed frequency is off).

Mod: Source and amount for resample frequency modulation.



The *Audio S/H* module has one audio output.

### Hilbert

The *Hilbert* module features two Hilbert transforms, a Sine/Cosine generator and a dual Ring modulator. These sections can be combined to setup a *Bode* frequency shifter. The module has the following parameters:

Hilbert Input A/B: Source of the input for the Hilbert transforms.

(Sine/Cosine Generator On/Off): Switches the Sine/Cosine generator on or off.

(Sine/Cos) Frequency: Fixed frequency of the Sine/Cosine generator.

(Sine/Cos Frequency) Mod: Source and amount for frequency modulation of the Sine/Cosine generator.

Ring Input A1/A2: Sources of the inputs for Ring modulator A.

Ring Input B1/B2: Sources of the inputs for Ring modulator B.



The *Hilbert* module has nine audio outputs (*Hilbert A Real*, *Hilbert A Imaginary*, *Hilbert B Real*, *Hilbert B Imaginary*, *Cosine*, *Sine*, *Ring A*, *Ring B*, *Ring A+B*).



#### Example: Frequency shifting using the Sine/Cosine generator

- Switch *Sine/Cosine Gen* to "on"
- Plug the signal to *Hilbert Input A*
- Plug *Hilbert A Real* to *Ring A1*
- Plug *Hilbert A Imag.* to *Ring A2*
- Plug *Cosine* to *Ring B1*
- Plug *Sine* to *Ring B2*
- Use the *Ring A+B* output

#### Example: Frequency shifting using another signal

- Switch *Sine/Cosine Gen* to "off"
- Plug the signal to *Hilbert Input A*
- Plug the other signal to *Hilbert Input B*
- Plug *Hilbert A Real* to *Ring A1*
- Plug *Hilbert A Imag.* to *Ring A2*
- Plug *Hilbert B Real* to *Ring B1*
- Plug *Hilbert B Imag.* to *Ring B2*
- Use the *Ring A+B* output



## Bit Reducer

The *Bit Reducer* module features two bit reducers with the following parameters:

Input A/B: Source and level of the input.

Scale A/B: Scale (pre-amplification) of the input signal.

Bits A/B: Number of (virtual) bits of reduction.

Mod A/B: Source and amount for bit reduction modulation.

The *Bit Reducer* module has two audio outputs (*A* and *B*).



## Dual Saturator

The *Dual Saturator* module features two saturation/distortion units with the following parameters:

Input A/B: Source and level of the input.

Drive A/B: Amount of saturation/distortion of the input signal.

(Drive) Mod A/B: Source and amount for saturation/distortion modulation.

Compensate A/B: Amount of output level compensation.

The *Dual Saturator* module has two audio outputs (*A* and *B*).





## Analyzer

The *Analyzer* module is a 10-stage resonator bank typically used as the analysis block of a vocoder in conjunction with the *Filter Bank*. It has the following parameters:

**Input:** Level and source of the left input signal.

**LAG:** Smoothing of the output modulation signals.

**(Sharpness):** Resolution of the resonators (*SOFT* or *SHARP*).

**Frequency Bands (A – J):** Frequency of the respective frequency band in Hertz.

The *Analyzer* module has ten modulation outputs (one per frequency band).



## How To Build A Simple Vocoder (Example):

A simple vocoder would use the polyphonic signal of an *Oscillator* voice module and feed it into a *Filter Bank* master module which is modulated by an *Analyzer* master module – the latter being fed with an external signal (see figure below).





## Modulator Modules

### Envelope

The *Envelope* module is an exponential ADSR generator with the following parameters:

Gate: Sources of the two gate/trigger inputs.

Attack: Attack time.

Decay: Decay time.

Sustain: Sustain level.

Release: Release time.

(Attack) Mod: Source and amount for Attack time modulation.

(Decay) Mod: Source and amount for Decay time modulation.

(Sustain) Mod: Source and amount for Sustain level modulation.

(Release) Mod: Source and amount for Release time modulation.



The *Envelope* module has one modulation output.

### Dual Envelope

The *Dual Envelope* module features two exponential ADSR generators with the following parameters:

Gate A/B: Source of the gate/trigger input.

Attack A/B: Attack time.

Decay A/B: Decay time.

Sustain A/B: Sustain level.

Release A/B: Release time.

The *Dual Envelope* module has two modulation outputs (*A* and *B*).



## Quad Envelope

The *Quad Envelope* module features four exponential AD generators with the following parameters:

Gate A/B/C/D: Source of the gate/trigger input.

Attack A/B/C/D: Attack time.

Decay A/B/C/D: Decay time.

The *Quad Envelope* module has four modulation outputs (A to D).



## Dual Linear Envelope

The *Dual Linear Envelope* module features two linear ADSR generators with the following parameters:

Gate A/B: Source of the gate/trigger input.

Attack A/B: Attack time.

Decay A/B: Decay time.

Sustain A/B: Sustain level.

Release A/B: Release time.

The *Dual Linear Envelope* module has two modulation outputs (A and B).



## LFO

The *LFO* module is a low frequency oscillator (LFO) with the following parameters:

Waveform: The waveform of the LFO (*Sinoid*, *Trapezoid*, *Pulse*, *Random*).

Shape: The shape of the waveform. For example a trapezoid with shape 0.00 results in a triangle waveform, while shape -1.00 results in a falling and +1.00 in a rising saw-like waveform.

Phase: The starting phase when the LFO is reset.

Rate: The LFO rate.

(Mod Rate): Source and amount of LFO rate modulation.

Depth: The LFO depth/intensity.

(Depth Mod): Source and amount of LFO depth modulation.

Sync: Rate of to-host synchronization.

Reset: Gate/trigger source for LFO reset.



The *LFO* module has two modulation outputs: *LFO* (bipolar) and *LFO+* (unipolar).

## Dual LFO

The *Dual LFO* module features two low frequency oscillators (LFOs) with the following parameters:

Waveform A/B: The waveform of the LFO (*Sine*, *Triangle*, *Saw Down*, *Saw Up*, *Square*, *Random*).

Rate A/B: The LFO rate.

Depth A/B: The LFO depth/intensity.

(Depth Mod A/B): Source and amount of LFO depth modulation.

Sync A/B: Rate of to-host synchronization.

Reset A/B: Gate/trigger source for LFO reset.

The *Dual LFO* module has four modulation outputs: *LFO A* (bipolar), *LFO A+* (unipolar), *LFO B* (bipolar), *LFO B+* (unipolar).



## Sample & Hold

The *Sample & Hold* module features two Sample & Hold units and an additional noise generator. It has the following parameters:

White: Level of the white noise.

Pink: Level of the pink noise.

Rate A/B: The Sample & Hold rate.

Trigger A/B: Trigger source for Sample & Hold (overrides *Rate*).

Input: The input source for Sample & Hold.

The *Sample & Hold* module has two audio outputs (*White Noise* and *Pink Noise*) and two modulation outputs (*A* and *B*).



## Sequencer Modules

### Sequencer

The *Sequencer* module is an eight-step analog-style sequencer with the following parameters:

Clock: Trigger source for the sequencer clock.

Reset: Trigger source for the sequencer reset.

Link: Modulation source for sequencer link; when linking the sequencer with another sequencer module, the output of the other module should be added to the *Link* input.

It works as follows: If the sequencer is in non-looping mode and has finished the last sequence step, then the output of the sequencer will be determined by the *Link* input.

(Loop): Switches the sequencer from looping to non-looping.

Steps: The number of active sequencer steps.

Step 1-8: The value of the respective sequencer step.



The *Sequencer* module has one modulation output and twelve trigger outputs: *Clock*, *Clock Link* (clocks only in non-looping mode when the sequencer has finished the last sequence step), *Reset* (clocks when the sequencer resets to step 1), *Stop*, *Step 1-8* (clocks for each individual step).

### Beat Sequencer

The *Beat Sequencer* module features four sixteen-step trigger sequencers with the following parameters:

Clock: Trigger source for the sequencer clock.

Reset: Trigger source for the sequencer reset.

Div: Clock division factor.

(Loop): Switches the sequencer from looping to non-looping.

Steps: The number of active sequencer steps.

Step 1-16 A/B/C/D: The status (on/off) of the respective sequencer step.



The *Beat Sequencer* module has nine trigger outputs: *Clock*, *Clock Div*, *Clock Link* (clocks only in non-looping mode when the sequencer has finished the last sequence step), *Reset* (clocks when the sequencer resets to step 1), *Stop*, *Line A-B* (clocks for the individual sequencer lines).

## Sequential Switch

The *Sequential Switch* module is an eight-step sequencer that switches between up to eight modulation inputs. It has the following parameters:

**Rate:** The sequencer rate.

**Clock:** Trigger source for the sequencer clock (overrides *Rate*).

**C.Div:** Clock division factor.

**Reset:** Trigger source for the sequencer reset.

**Link:** Modulation source for sequencer link; when linking the sequencer with another sequencer module, the output of the other module should be added to the *Link* input.

It works as follows: If the sequencer is in non-looping mode and has finished the last sequence step, then the output of the sequencer will be determined by the *Link* input.

**(Loop):** Switches the sequencer from looping to non-looping.

**Steps:** The number of active sequencer steps.

**In 1-8:** The modulation source inputs of the respective step.



The *Sequential Switch* module has one modulation output and fourteen trigger outputs: *Clock*, *Clock Div*, *Clock Link* (clocks only in non-looping mode when the sequencer has finished the last sequence step), *Reset* (clocks when the sequencer resets to step 1), *Stop*, *Step Active* (clocks each time a step with active modulation input is reached), *Step 1-8* (clocks for each individual step).

## Audio Switch

The *Audio Switch* module is an eight-step sequencer that switches between up to eight audio inputs. It is identical to the *Sequential Switch* module with the difference that it switches audio signals instead of modulation signals.



## Sequencer Clock

The *Sequencer Clock* module features a clock trigger source, two gate/trigger signal switches, a gate/trigger signal join, and two Modulation-to-Gate converters. It has the following parameters:

Rate: The clock rate.

(Clock On/Off): Switch and source for enabling the clock signal.

Switch 1/2: Source of a gate/trigger signal to enable/disable.

(Switch 1/2 On/Off): Switch and source for enabling the respective gate/trigger signal.

(Join Input 1-3): Source of the Join input signal.

(M2G Input A/B): Source of the M2G input signal.

(M2G Threshold A/B): Threshold of the M2G. If the input signal is higher than the threshold, the gate output of the M2G will be active, else it will be inactive.



The *Sequencer Clock* module has six gate/trigger outputs (*Clock*, *Switch 1*, *Switch 2*, *Join*, *M2G A*, *M2G B*).



## Effect Modules

### Phaser

The *Phaser* module is a 4-stage Phaser effect with integrated LFO. It has the following parameters:

Input A/B: Level and source of the input signal.

(Mono/Stereo): Switch from mono (A+B) to stereo input (A|B).

Center: Center frequency of the Phaser in Hertz.

Separation: Separation of the frequencies of the four stages.

Feedback: Feedback amount.

(Frequency Mod): Source and amount for center frequency modulation.

Mix: Mix between original and effect signal.

Rate: The LFO rate.

Depth: The LFO depth/intensity.



The *Phaser* module has two audio and two modulation (LFO) outputs.

### BBD Delay / Stereo BBD Delay

The *BBD Delay* and *Stereo BBD Delay* modules simulate a (stereo) bucket brigade delay with the following parameters:

Input A/B: Level and source of the input signal.

Time: Delay time.

(Time Range): Range of the delay time.

(Time) Mod: Source and amount for delay time modulation.

Freeze: Freezes the delay signal.

(Freeze Trigger): Trigger signals to freeze the delay signal.

Feedback: Feedback amount.

Mix: Mix between original and effect signal.

Quality: The quality of the effect signal.



The *BBD Delay* module has one while the *Stereo BBD Delay* module has two audio outputs.

## Dual Delay Line

The *Dual Delay Line* module features two delay lines with the following parameters:

Input A/B: Level and source of the input signal.

Delay A/B: Delay time.

Clear A/B: Gate/trigger source for clearing the delay line.

Feed Forward: Feed forward coefficient of the delay line.

Feed Back: Feed back coefficient of the delay line.

Setting the *Feed Forward* coefficient to the exact negative value of the *Feed Back* coefficient will result in an Allpass filter delay.

The *Dual Delay Line* module has two audio outputs (*A* and *B*).



## Reverb

The *Reverb* module simulates a stereo reverb with the following parameters:

Input A/B: Level and source of the input signal.

(Mono/Stereo): Switch from mono (*A+B*) to stereo input (*A|B*).

Predelay: Predelay time.

Size: Size of the reverberation room.

Damp: Damping of the reverb.

Length: Length of the reverb.

Spread: Stereo spread of the reverb signal.

Mix: Mix between original and effect signal.

The *Reverb* module has two audio outputs (*Left* and *Right*).



## Tool Modules

### Voltage Processor

The *Voltage Processor* module features four modulation signal processors with the following parameters:

Input A/B/C/D: Source of the input signal.

Scale A/B: Amplitude of the input signal.

Offset A/B: Constant offset added to the input signal.

Mod C/D: Source and amount for amplitude modulation of the input signal.

The *Voltage Processor* module has four modulation outputs (*Processor A to D*).



### LAG/Adder

The *LAG/Adder* module features two LAG generators and two adders for modulation signals with the following parameters:

Input A/B: Source of the LAG generator input signal.

Rate A/B: Rate of the LAG generator.

(Adder Input C1/C2): Sources and levels of the *Adder C* inputs.

(Adder Input D1/D2): Sources and levels of the *Adder D* inputs.

The *LAG/Adder* module has four modulation outputs (*LAG A, LAG B, Adder C, Adder D*).



## Sample Delay

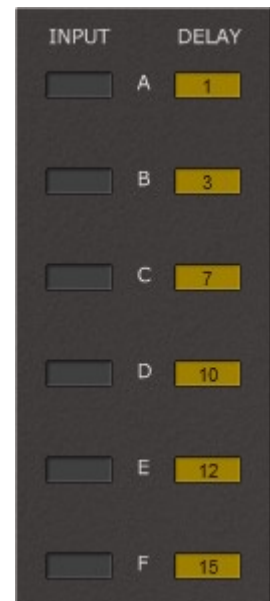
The *Sample Delay* module features six sample delays with the following parameters:

Input A-F: Source of the input signal.

Delay A-F: Delay of the input signal (1 to 15 samples).

Since each module of *ModulAir* delays its input signal by one sample, the *Sample Delay* is useful for compensating the delay between different signal paths. For example if you mix two audio signals *A* and *B* where *B* is routed through an extra module (e.g. a filter), the signal “phase” between *A* and *B* will differ by one sample. If you pass signal *A* through a *Sample Delay*, the resulting signals will be “in sync” again.

The *Sample Delay* module has six audio outputs (*Delay A* to *F*).



## Modulation Signal Delay/M2G

The *Modulation Signal Delay/M2G* module features a modulation signal delay, a modulation signal adder and two Modulation-to-Gate converters. It has the following parameters:

(Delay) Input A/B: Source of the delay input signal.

Delay: Delay time.

Feedback: Feedback of the delay.

(Adder Input 1/2): Sources and levels of the adder inputs.

(M2G Input A/B): Source of the M2G input signal.

(M2G Threshold A/B): Threshold of the M2G. If the input signal is higher than the threshold, the gate output of the M2G will be active, else it will be inactive.

The *Modulation Signal Delay/M2G* module has two modulation outputs (*Delay*, *Adder*) and two gate/trigger outputs (*M2G A*, *M2G B*).



## Gate Tools

The *Gate Tools* module features two gate modifiers, two trigger dividers and a gate/trigger signal join. It has the following parameters:

(Modifier Input A/B): Source of the Modifier input signal.

Length A/B: Length of the modified gate signal.

(Length) Mod A/B: Source and amount for length modulation of the modified gate signal.

(Divide Input C/D): Source of the Divider input signal.

(Divisor C/D): Divisor of the Divider.

(Join Input 1-4): Source of the Join input signal.

The *Gate Tools* module has five gate/trigger outputs (*Gate A*, *Gate B*, *Divider C*, *Divider D*, *Join E*).



## Oscilloscope

The *Oscilloscope* is a simple tool for viewing audio or modulation signals. It has the following parameters:

Input A: Audio or modulation signal input A. This input is also used for triggering the oscilloscope.

Input B: Audio or modulation signal input B.

(Mode): In *A+B* mode the two input signals A and B are added while in *XY* mode input A drives the horizontal and input B the vertical axis of the display (great for creating *Lissajous* curves).

Time: Time resolution (0.5 milliseconds to 2 seconds).

Trig.: Trigger level of signal A. Note that the trigger bypasses the attenuation of input A – this way you can trigger the oscilloscope even if the input level knob is set to 0.

(Trigger Mode): Trigger on raising or falling edge of signal A.



### Keyed Gate, Keyed Modulation, Keyed Signal

The *Keyed Gate*, *Keyed Modulation* and *Keyed Signal* modules can be used to restrict an incoming gate, modulation or audio signal to a range of keys.

IN A-H: Selects the input signal for the respective channel.

MIN A-H: Sets the lowest key for the respective channel.

MAX A-H: Sets the highest key for the respective channel.

The *Keyed Gate* module has eight outputs (one for each channel). The *Keyed Modulation* and *Keyed Signal* modules feature an additional summed output.



## Miscellaneous Functionality

### Options Menu

When clicking on the *Menu* button, a context menu opens with the following options:

<b>Copy Program</b>	Copy current program to internal clipboard
<b>Paste Program</b>	Paste internal clipboard to current program
<b>Init Program</b>	Initialize the current program
<b>Load Program</b>	Load a program file containing a patch to <i>ModulAir's</i> current program
<b>Save Program</b>	Save <i>ModulAir's</i> current program to a program file
<b>Load Bank</b>	Load a bank file containing 64 <i>ModulAir</i> patches
<b>Save Bank</b>	Save <i>ModulAir's</i> 64 patches to a bank file
<b>Select Startup Bank</b>	Select the bank file that should always be loaded when <i>ModulAir</i> is started
<b>Load Startup Bank</b>	Load the Startup bank file; can also be used to check what the current Startup bank is
<b>Unselect Startup Bank</b>	Unselect the current Startup bank
<b>MIDI Thru</b>	Set globally if MIDI data sent to <i>ModulAir</i> should be sent through to its MIDI output
<b>Ignore Program Change</b>	Set globally if MIDI Program Change data sent to <i>ModulAir</i> should be ignored
<b>Highlight Sources</b>	Check to activate source highlighting
<b>Default Path for WAV Files</b>	Sets the default path for WAV files (for wave tables and samples)
<b>Visit fullbucket.de</b>	Open fullbucket.de in your standard browser

### MIDI Learn

Assigning MIDI controllers to *ModulAir* parameters is done via the MIDI Learn function. To activate *MIDI Learn*, click on the respective button and wiggle both the MIDI controller and *ModulAir's* parameter that you want to link. If you want to unlearn the assignment, right-click the *MIDI Learn* button (the label now reads "UNLEARN") and activate it. Now wiggle the MIDI controller or the parameter that you want to unlearn.

### Custom Tuning

For micro-tonal playing, you can load a TUN or SCL and KBM micro-tuning definition file by clicking on the Custom Tuning *FILE* button. Once you have loaded such a file you can switch micro-tuning on and off. For more information about micro-tuning



check the **Microtonal Synthesis** website at <http://www.microtonal-synthesis.com/> and Mark Henning's website at <https://www.mark-henning.de/>.

## **MIDI CC Sources**

16 MIDI controller signals can be used as modulation sources. To assign a MIDI CC to one of the 16 sources press the *SET* button and choose the respective controller. The same way works to unassign a controller.

## Frequently Asked Questions

### ***How do I install ModulAir (Windows 32 bit version)?***

Just copy the files `modulair.dll` from the ZIP archive you have downloaded to your system's or favorite DAW's VST plug-in folder. Your DAW should automatically register the *ModulAir* VST plug-in the next time you start it.

### ***How do I install ModulAir (Windows 64 bit version)?***

Just copy the file `modulair64.dll` from the ZIP archive you have downloaded to your system's or favorite DAW's VST plug-in folder. Your DAW should automatically register the *ModulAir* VST plug-in the next time you start it.

Note: You may have to remove any existing (32 bit) `modulair.dll` from your VST plug-in folder or else your DAW may screw the versions up...

### ***How do I install ModulAir (Mac VST/AU universal 32/64 bit)?***

Locate the downloaded PKG package file `modulair_1_2_0_mac.pkg` in Finder (!) and do a right- or control-click on it. In the context menu, click on "Open". You will be asked if you really want to install the package because it comes from an "unidentified developer" (me 😊). Click "OK" and follow the installation instructions.

### ***What is the plug-in ID of the ModulAir?***

The ID is fbMD.

### ***Will you support ModulAir?***

Yes. If you have problems, found a bug, or have some suggestions about *ModulAir* please send me a mail: [full.bucket@gmx.net](mailto:full.bucket@gmx.net) .

### ***How do I know if a new version of ModulAir is available?***

When connected to the Internet, open the Options menu (see section →*Options Menu*) by clicking the disk icon and select the entry "Check Online for Updates". If a new version of *ModulAir* is available on [fullbucket.de](http://fullbucket.de) the respective information will be shown in a message box.

### ***It crashes...!***

You have been warned: →*Introduction*

### ***I get no sound!***

Have you checked that a signal is connected to the overall output plugs in the →*Common Section*?