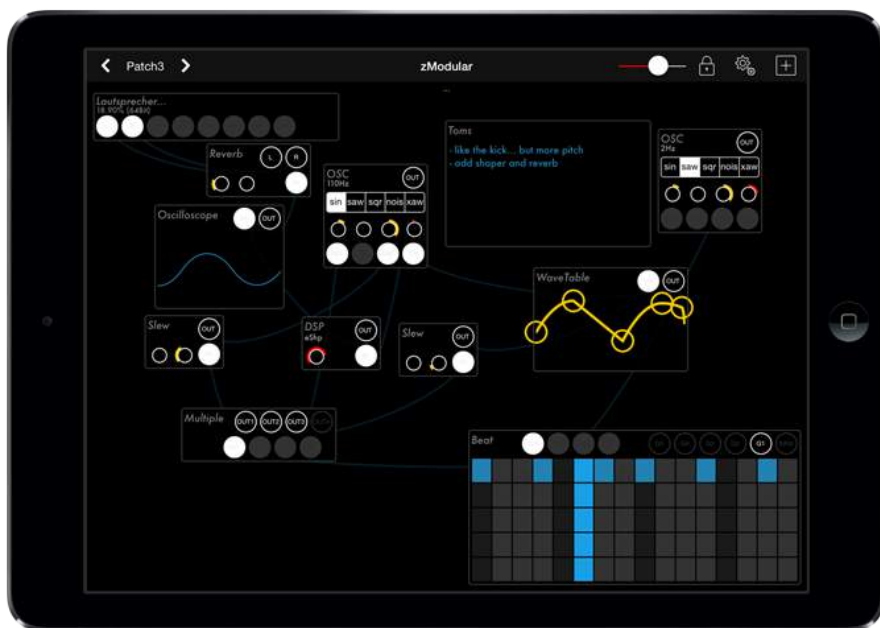




***zMors Modular*** is a modular synthesizer for the iPad

You can add various modules to your patches connecting them with cables

You can use hardware gear with up to 8 usb inputs/outputs for audio and/or control voltage send and MIDI in/out





# Table of Contents

---

Key Features.....	3
Getting Started.....	4
The Settings Menu.....	8
Exploring The WaveTable & Sampler Modules.....	16
THE GUI Options.....	20
Exploring The AU & IAA Modules.....	24
Exploring The Seq & CV Modules.....	31
Creating A Macro Module.....	39
THE MODULES.....	43
1 - GENERATORS.....	43
2 - ENVELOPES.....	45
3 - PROCESSORS.....	46
4 - TOOLS.....	48
5 - SEQ & CV.....	50
6 - MIDI.....	53
7 - FX.....	54
8 - GUI.....	56

## Key Features

64-bit support  
Input/output up to 8 audio/cv to your Modular Synth via USB Audio  
Optimized for use with Eurorack Synthesizers  
32bit Float Audio Engine with up to 16x oversampling  
Ableton Link Sync  
AudioUnit  
InterAppAudio up to 96K  
MIDI Input/Output  
Very Low Storage for App and Patches  
Skins with Custom Background Images

## Modules

OSC, Multimode Filter, WaveTable, ADSR, Slew, VCA,  
DSP, Combiner, Oscilloscope, Chords, CV Sequencer,  
Step Sequencer, Midi Keyboard, XY Pad, Delay,  
Reverb, BPM Clock, 4x4 Matrix Mixer,  
Macro for complex setups, Quantizer, Sample/Hold,  
AudioUnit Host (fx+synth), gate/cv to midi, 4 Band PEQ

## Tested & Working with External Hardware

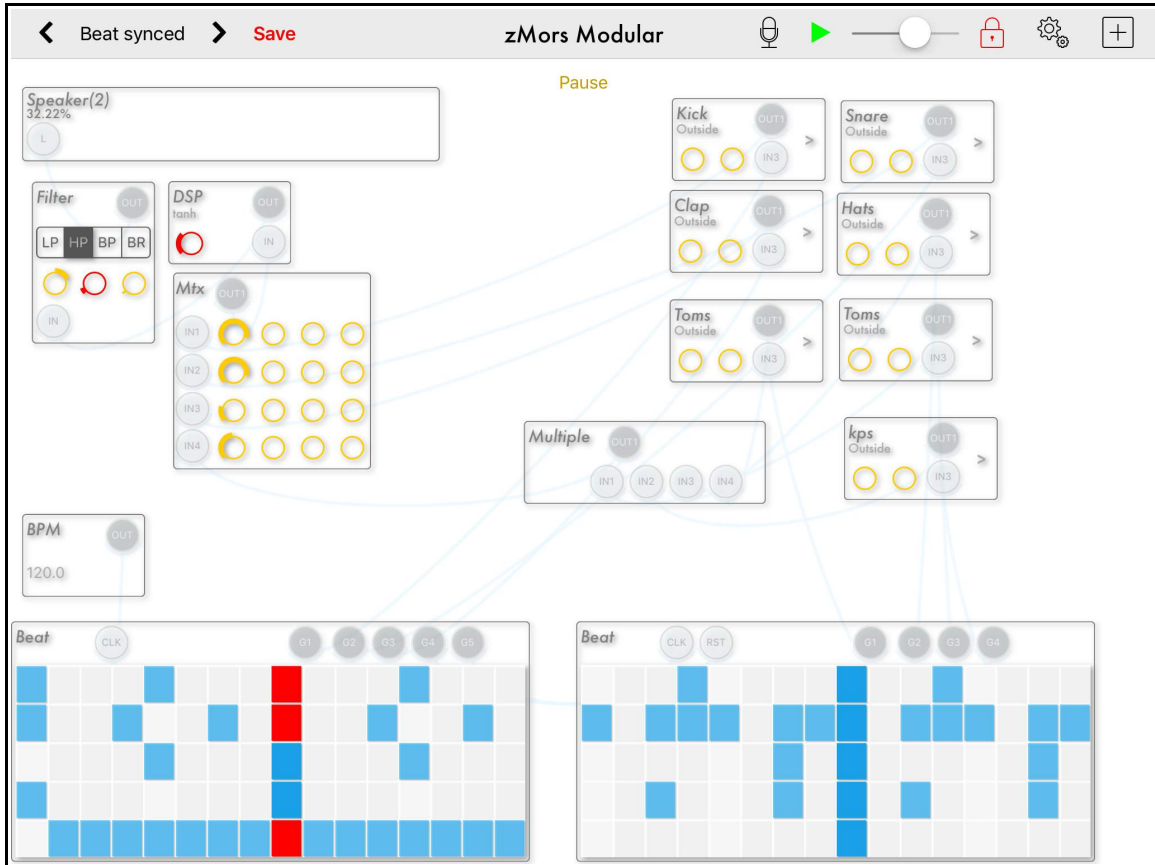
- Expert Sleepers ES-8
- AudioDamage Odio
- Endorphin.es - Shuttle Control
- Scarlett 18i20 and Expert Sleepers ES3
- RME Babyface and Expert Sleepers ES3
- MiniDSP USB Streamer Box and Expert Sleepers ES3/ES6

## FAQ

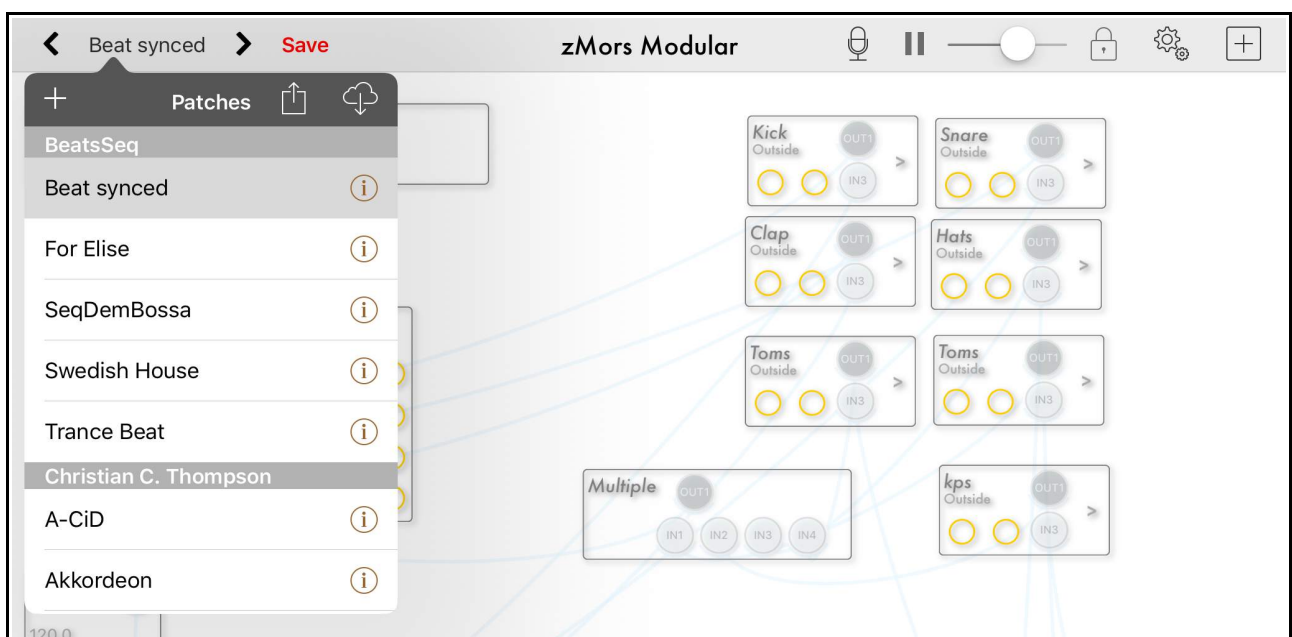
- The pulse trigger from the Ableton Link module can be setup in module settings
- Modules will only be rendered, when inserted in a chain connected to main output  
e.g. the oscilloscope
  - All preset will be stored in user document folder (json format),  
accessible by iTunes file sharing
    - The filenames are defined by {section}#{name}.json
- Tap on connected output ports to setup offset, level, etc per cable !!!!

# Getting Started

## The zMors Modular Interface



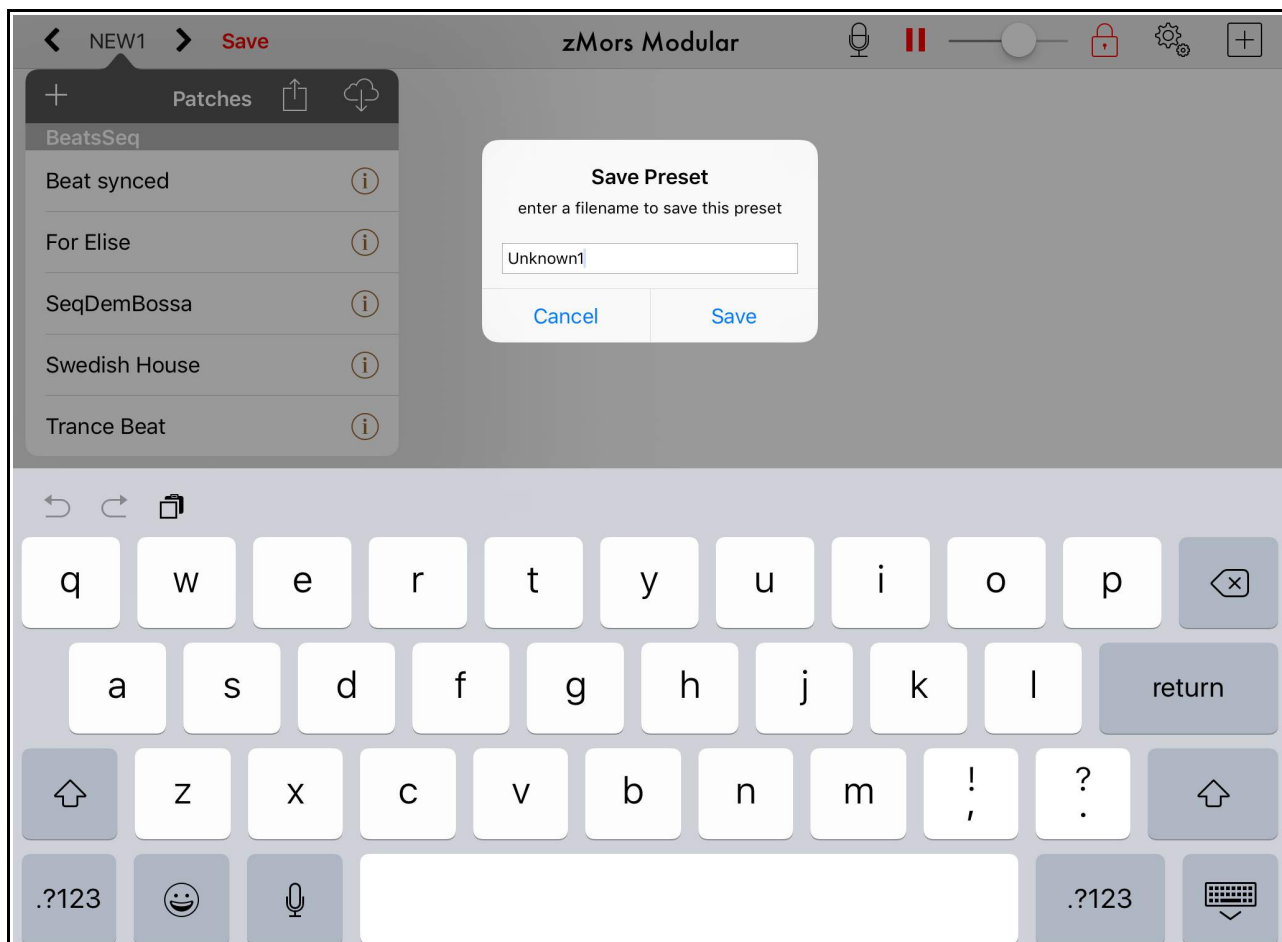
**Tap** on the patch name between the two arrows to access the **Patch List**.



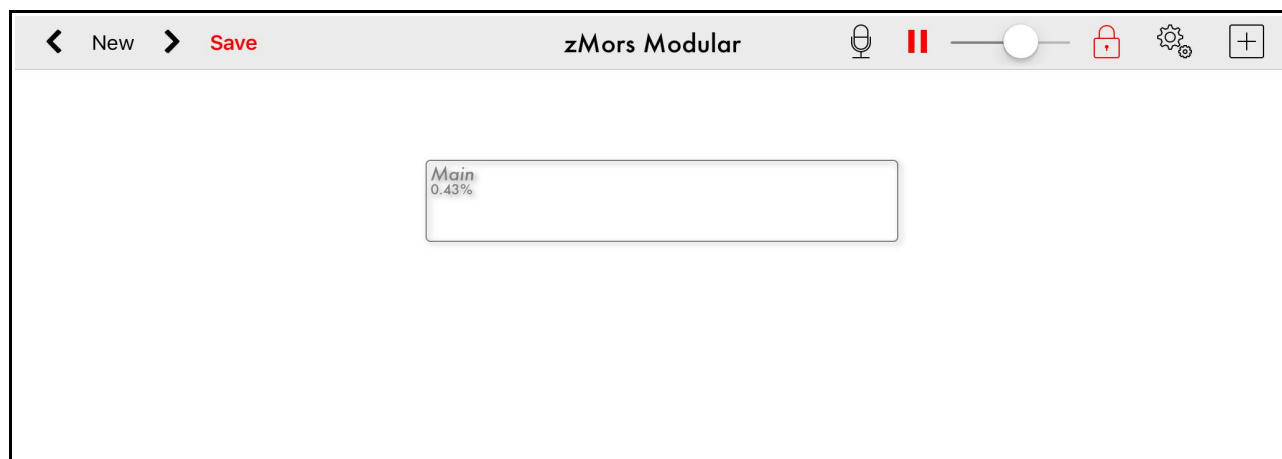
Tap on the + icon to create a **New Patch**.

Next you will be prompted to name your new patch.

For this example, I'll name the patch 'NEW'.

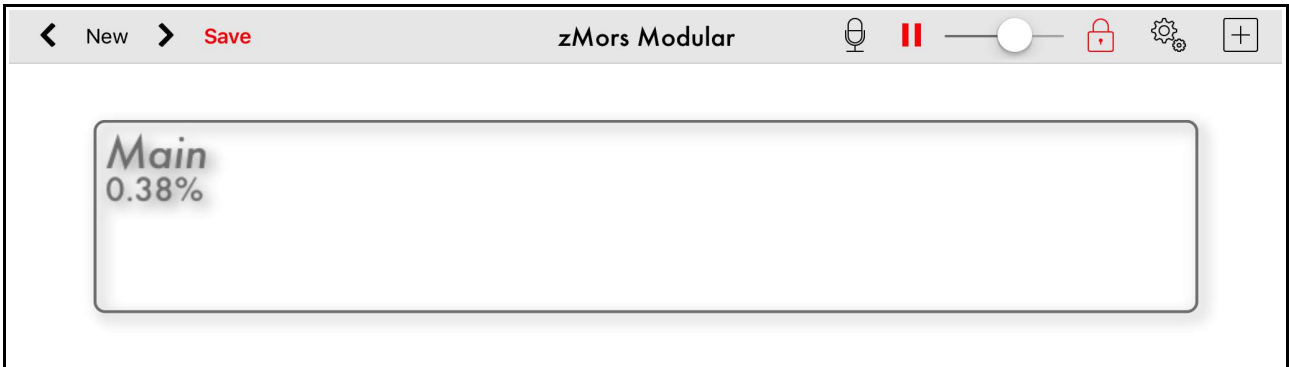


This next screen is what you are presented with after tapping the **Save** option above.

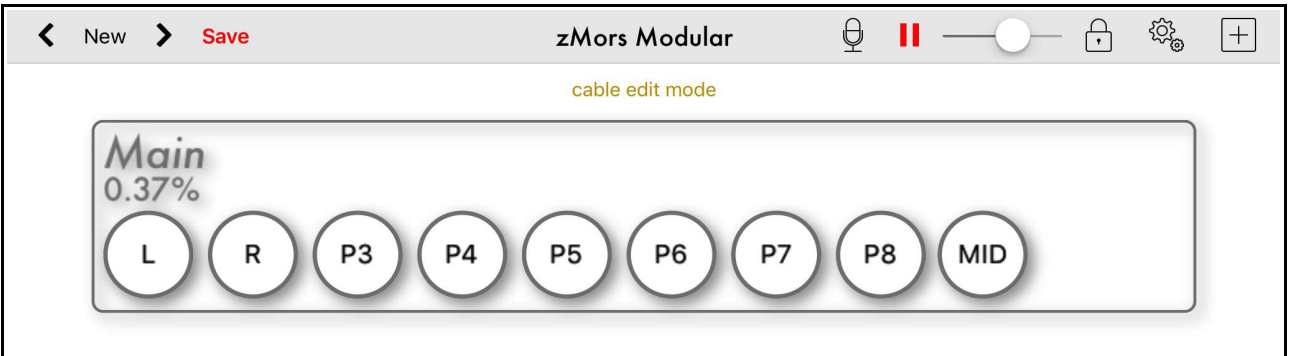


## Unlocking the cables

The **red lock** icon signifies that the cables are **locked**.



Simply tap on the red lock icon to **unlock** the cables access for modifying patches.



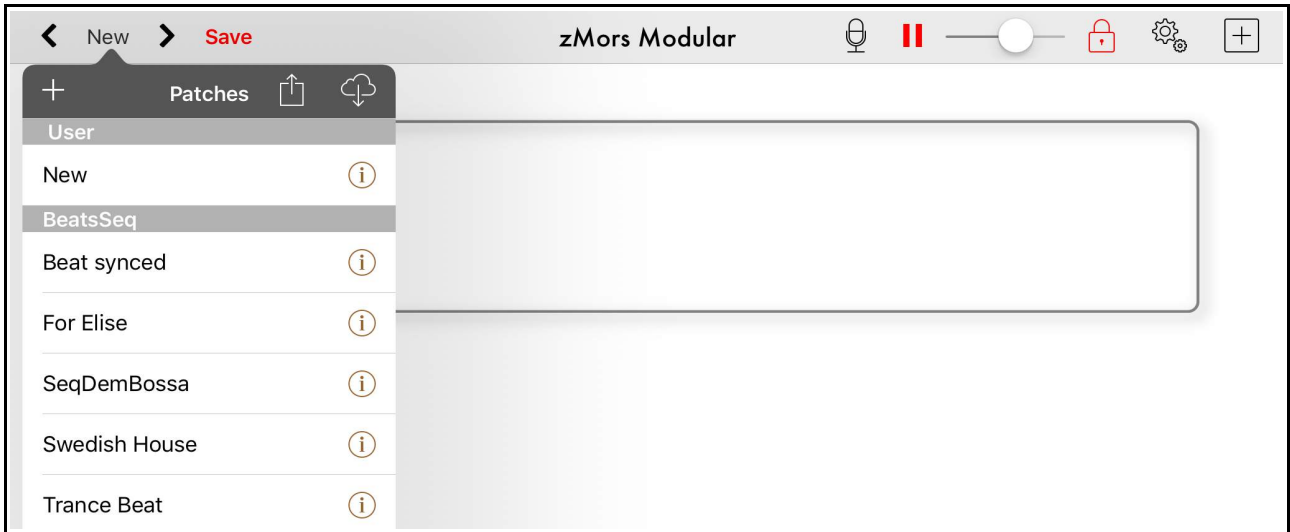
The lock icon will now appear grey, signifying that you have entered cable edit mode.



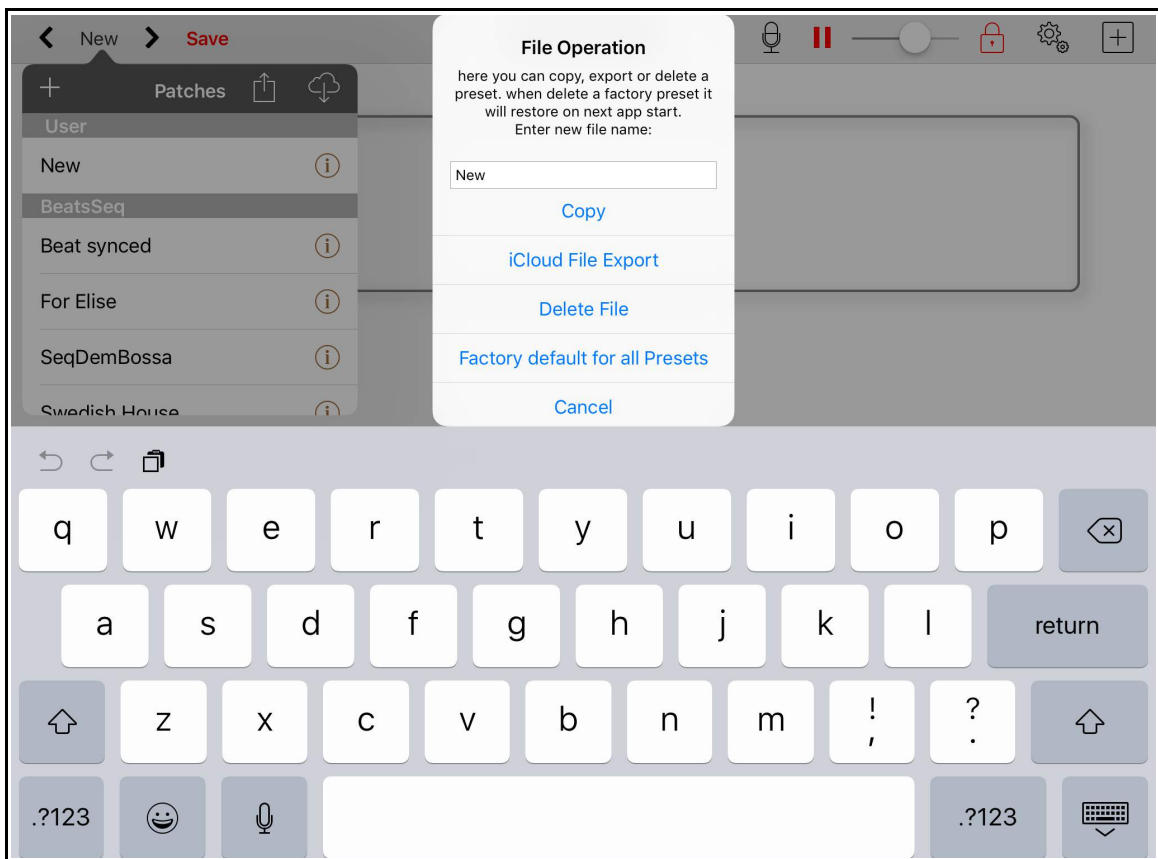
## Copying, Exporting & Deleting Presets

First, **tap** on the area between the two arrows to select a preset.





Next, **tap** on the **i** icon to access the file sharing options.



**Please note** - You can also use the **cloud** icon to access the file sharing options.

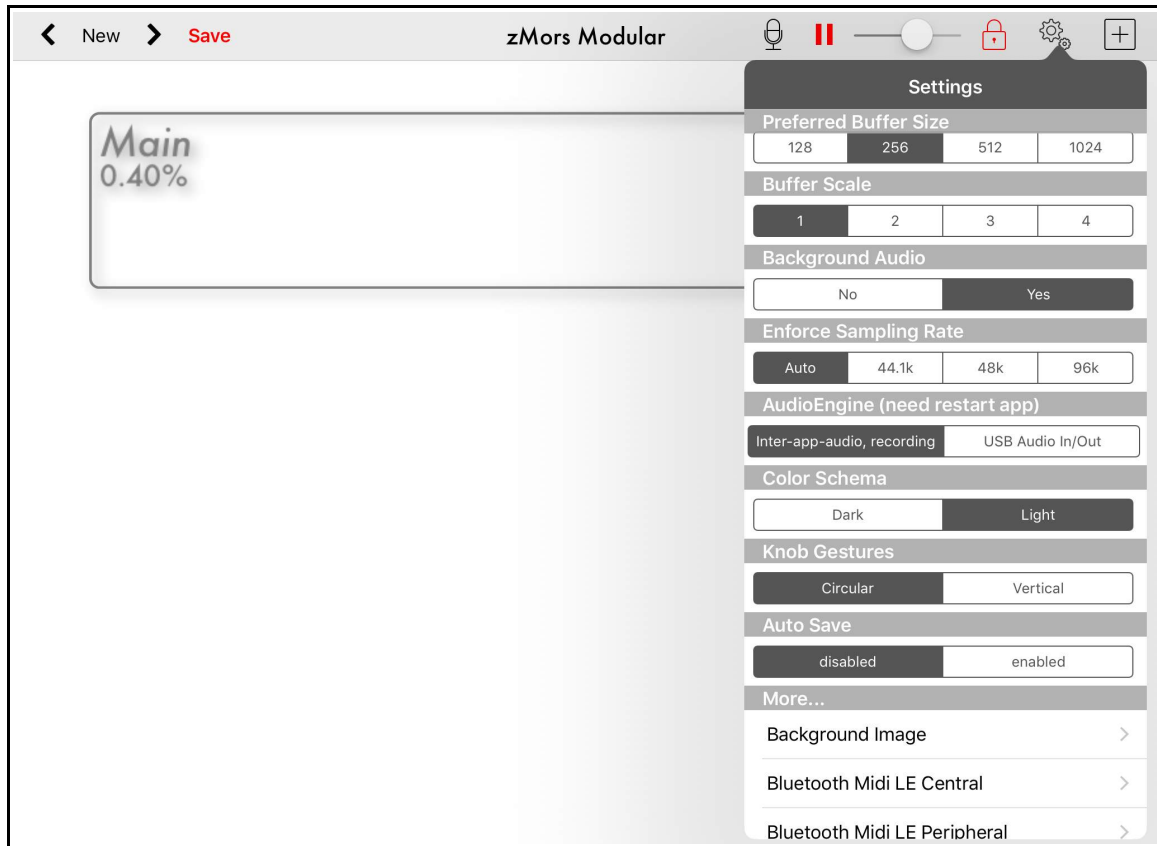


# The Settings Menu

Tap on the gear icon to access the settings menu.



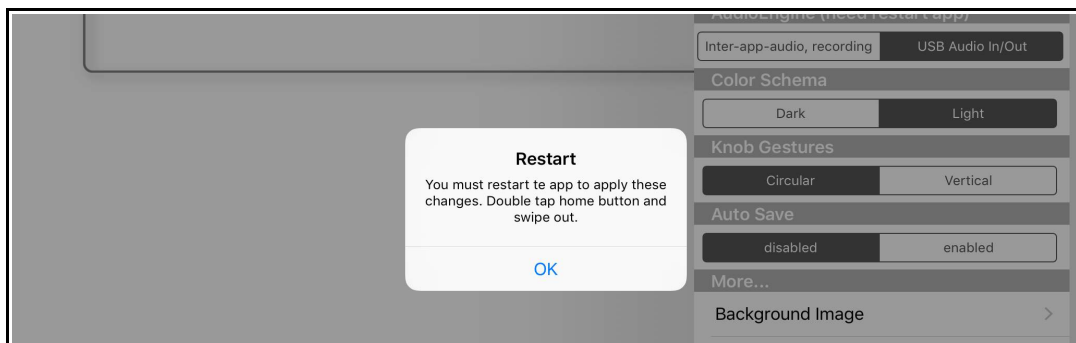
The settings menu gives you access to various parameters, including background audio, auto save, the latency buffer size and scale, as well the recording sample rate.



There are **2 audio engines** available -

- **USB Audio In/Out** for use with external hardware
- **InterAppAudio** with the **recording** function

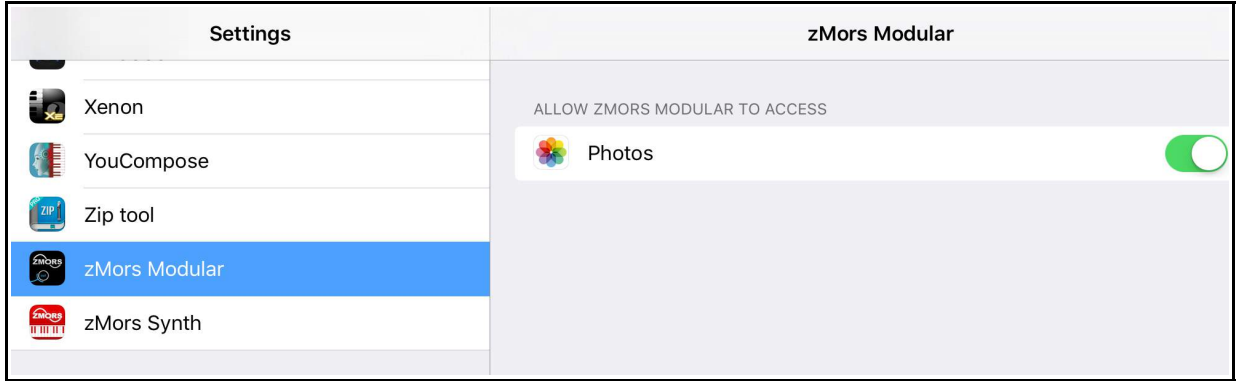
**Please note** - if you change the audio engine, you will need to restart the app again for the new engine to be active.



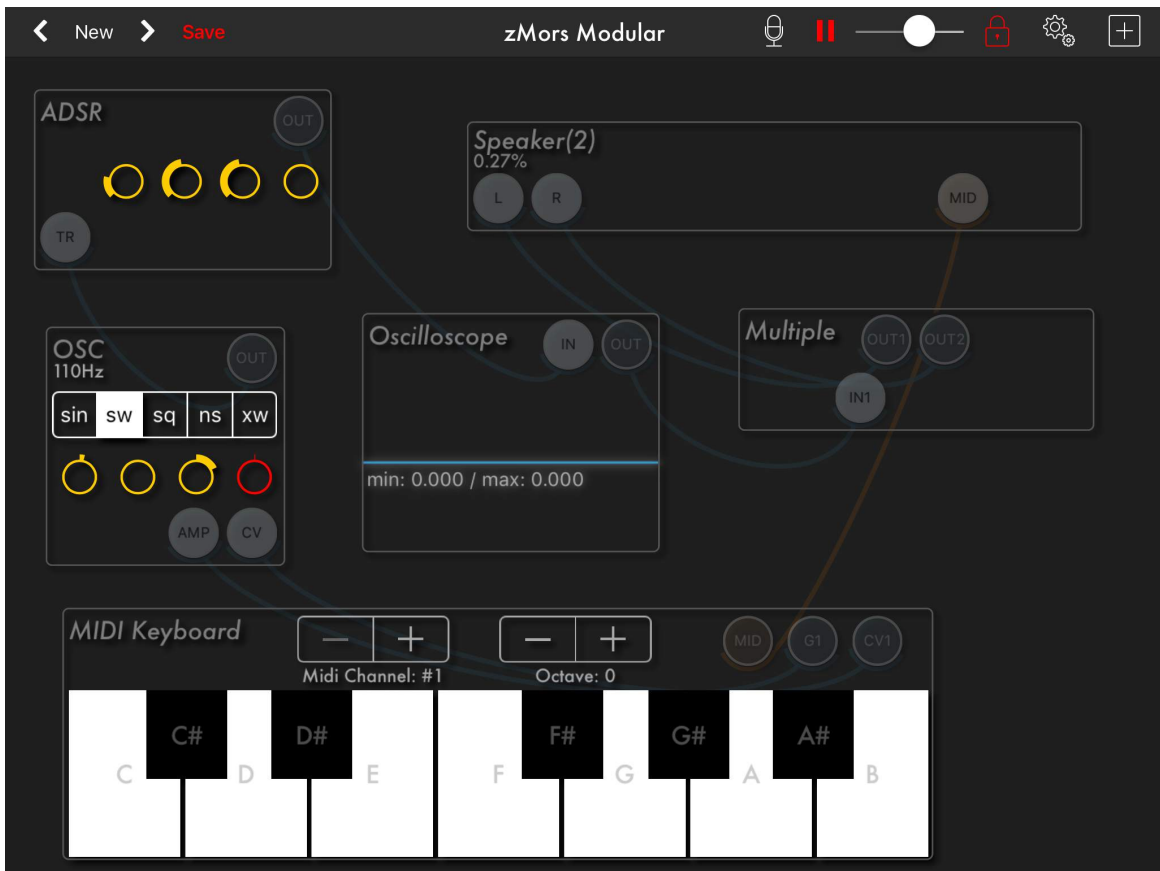


The Background Image may be changed to a photo from your Photo Album.

If you have trouble accessing your Photo Album for the background picture, simply go to your **iPads Settings Menu** and switch the access mode to on.

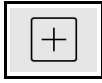


You can also change the **color scheme** to **Dark** mode.



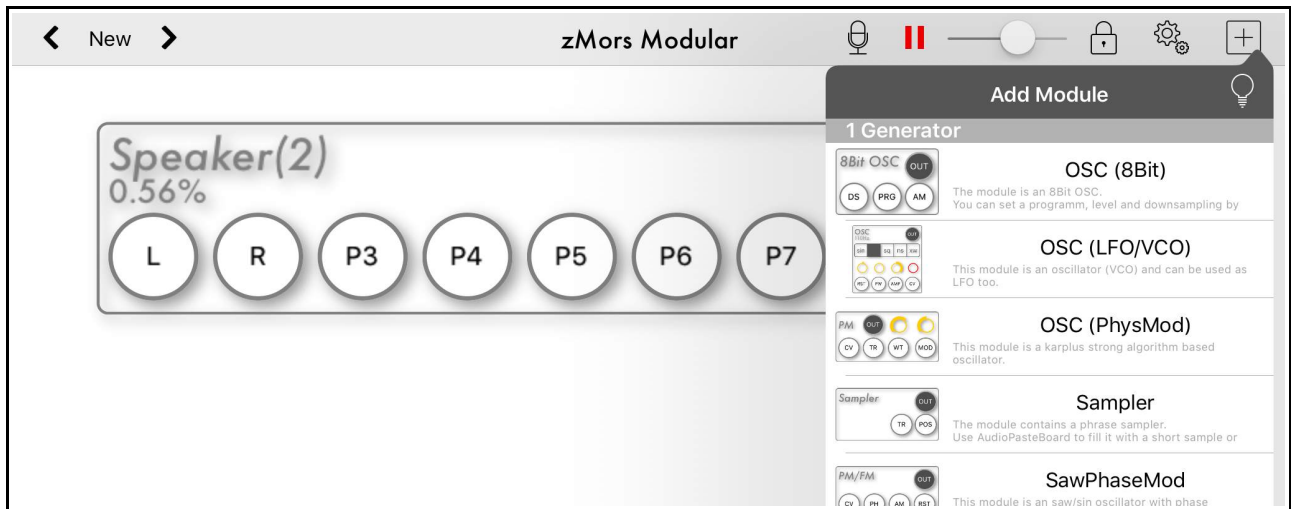
## Adding A Module

Tap on the **+** icon on the Top Bar to access the module list.

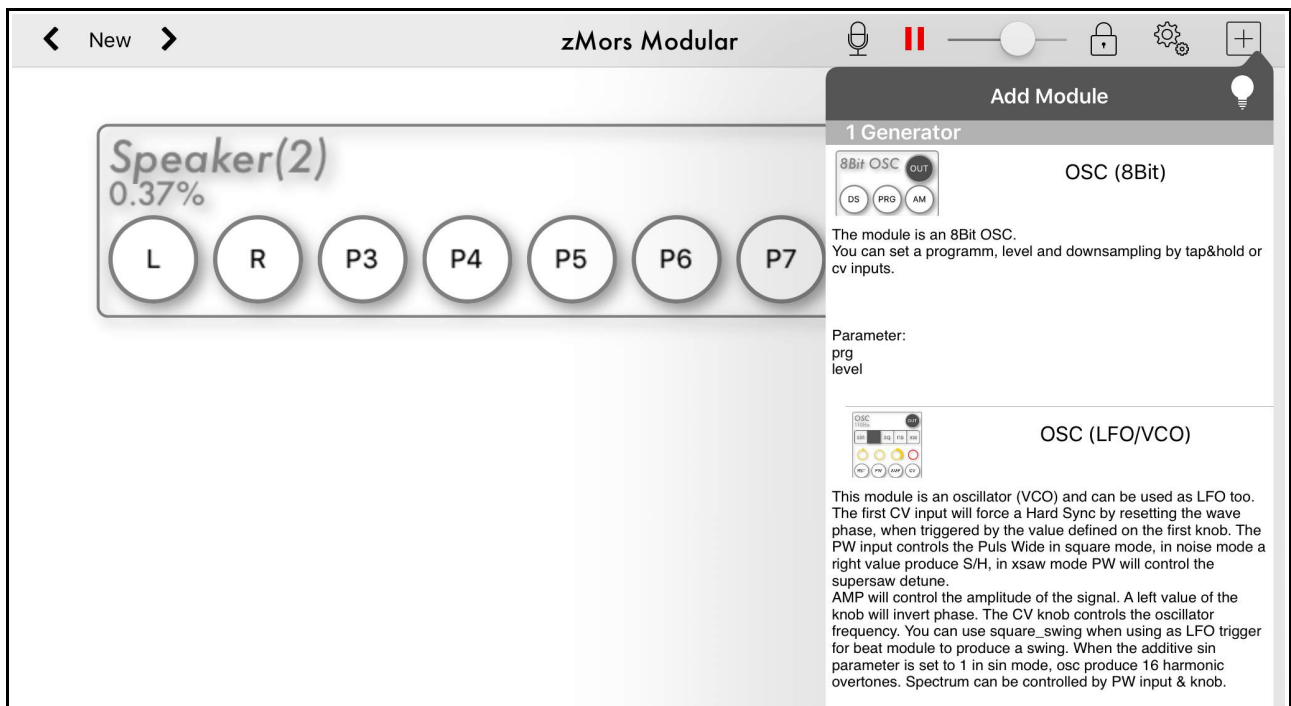


Next, tap on a module to add to your modular system.

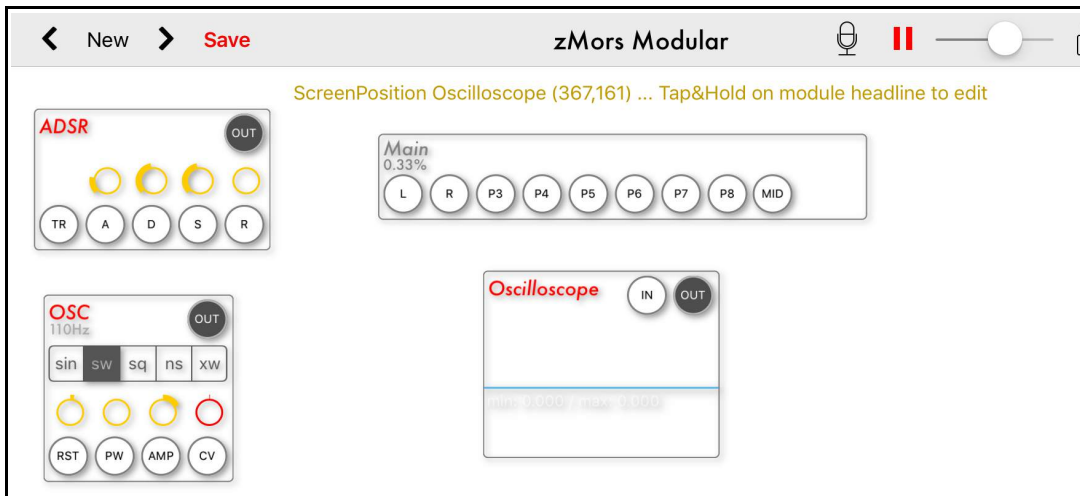
**Please note** - Expand these module titles by tapping on the lightbulb icon.



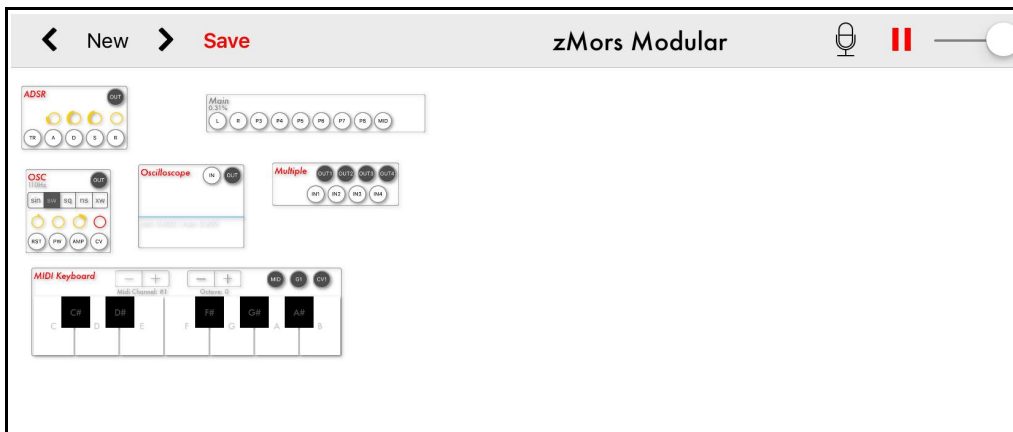
Notice the lightbulb icon is now lit and there is a detailed description for each module.



After you have added some modules, you can move them around using the **tap and drag** method.

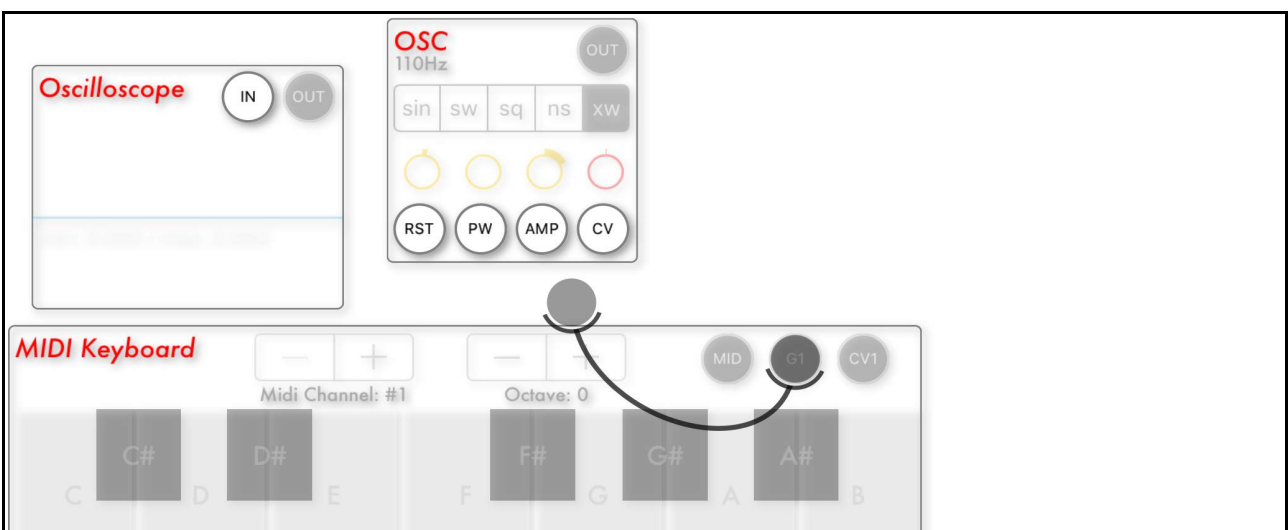


Use a **pinching motion** to reduce the size of your modules on the screen.



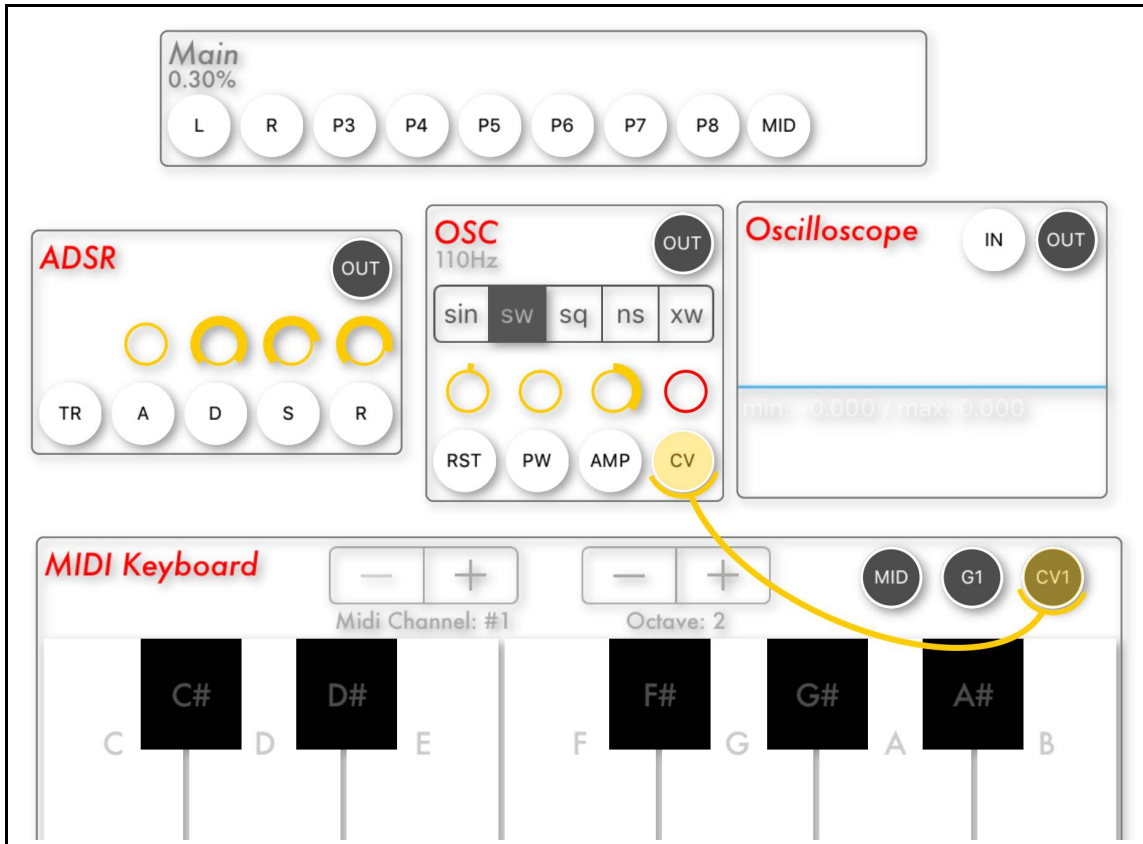
To start connecting the cables, simply **tap and drag** from one port (circle) to another.

The example below shows a cable being connected from the gate port (GT) of the keyboard, and being dragged towards the AMP port of the oscillator (OSC).

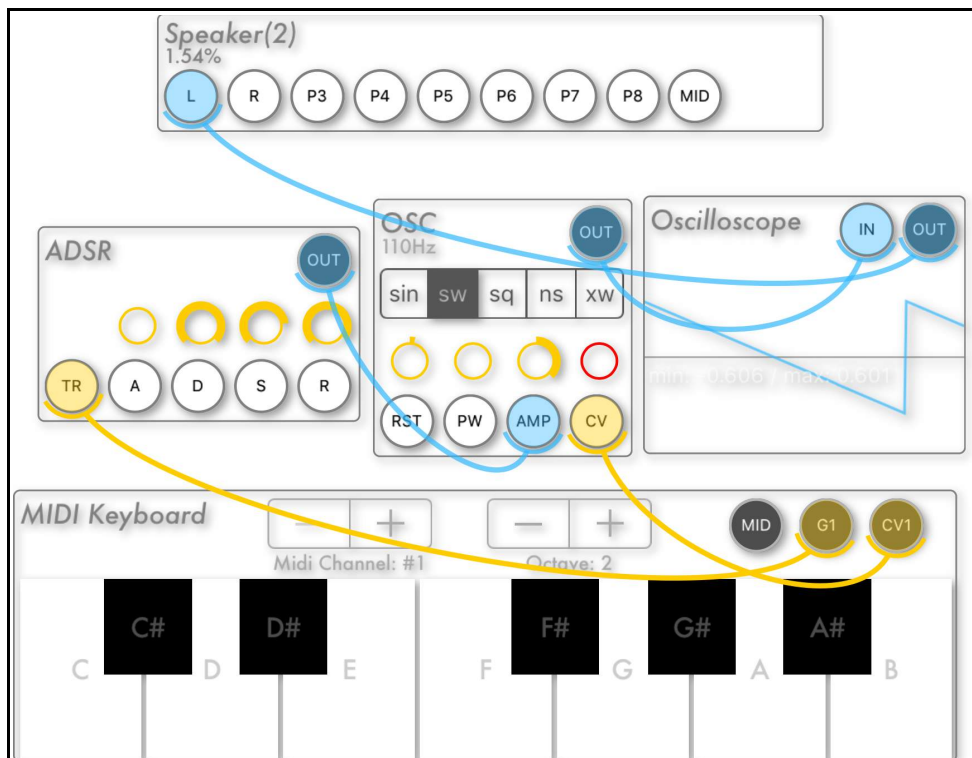




**Please note** - if you are using a keyboard, don't forget to connect up the **CV ports** to enable different notes to come from each key.

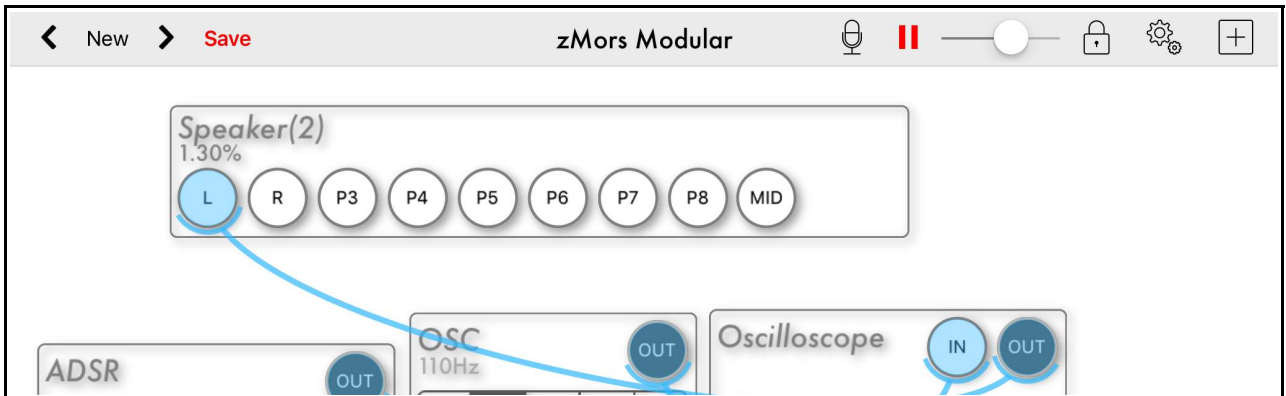


Below is an example of modules connected up with cables. It is a simple single oscillator synth with an ADSR envelope, playable from it's own keyboard.

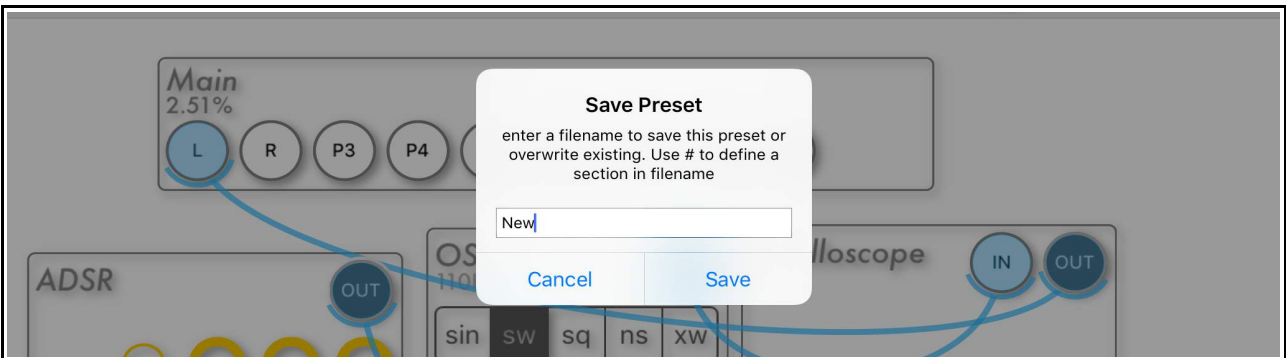


Now is a good time to save your new creation.

To do this, simply **tap** on **Save**.



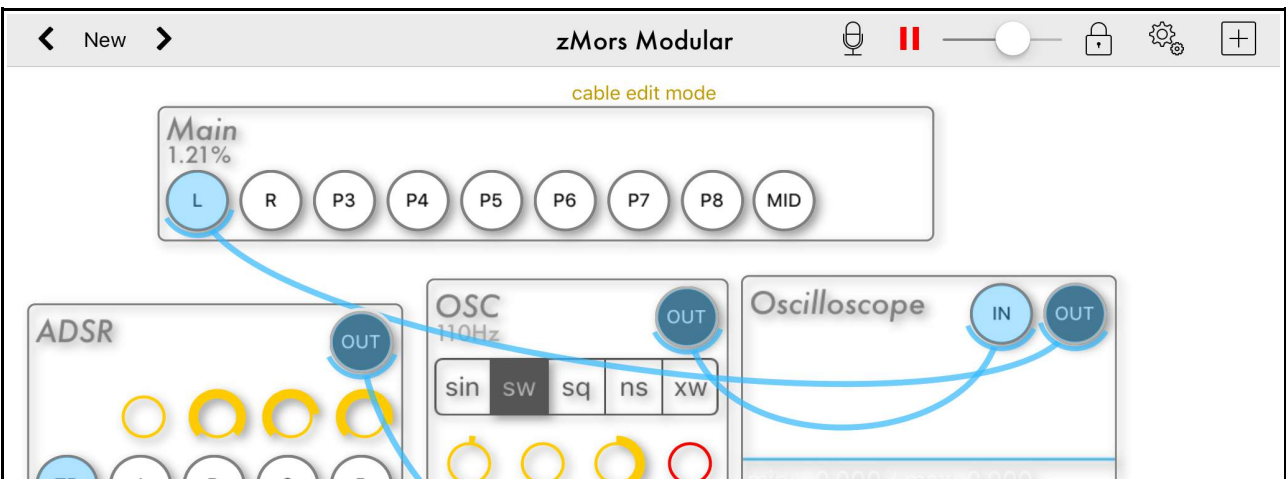
Next, name your preset and **tap** on the **Save** option.



**Please note** - you can set the saving behavior in the **Settings Menu** using **Auto Save**.

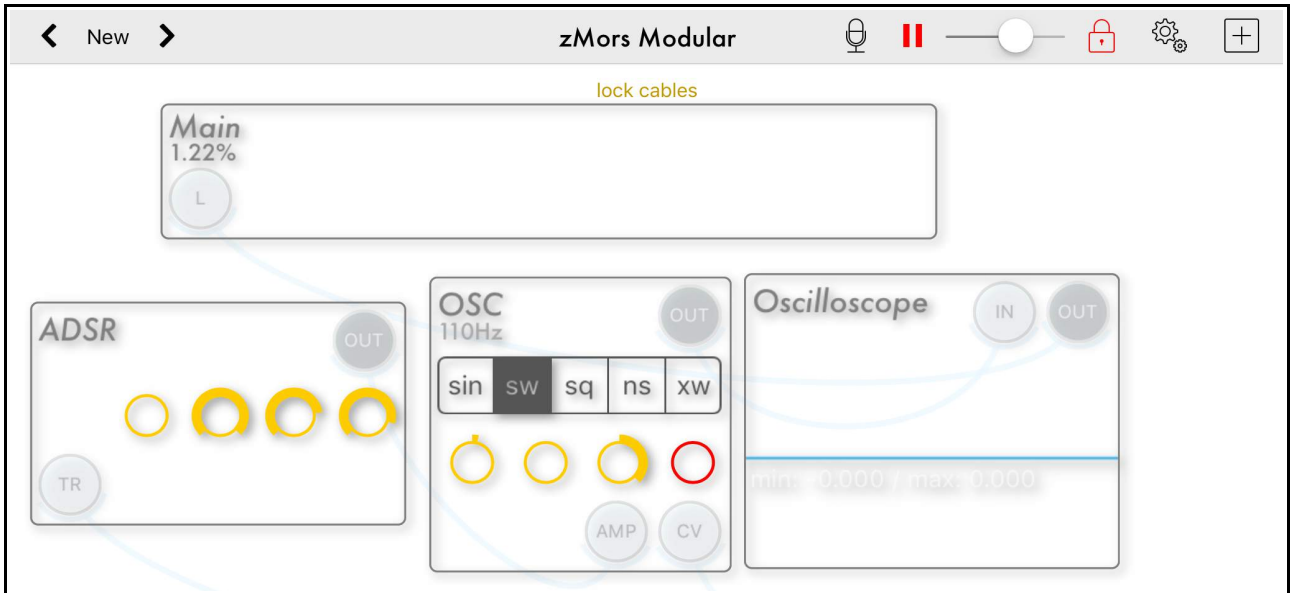
Now, **Tap** on the **lock** icon to access the **cable edit mode**.

The **lock** icon will turn **grey**, and you are able to edit your patch.

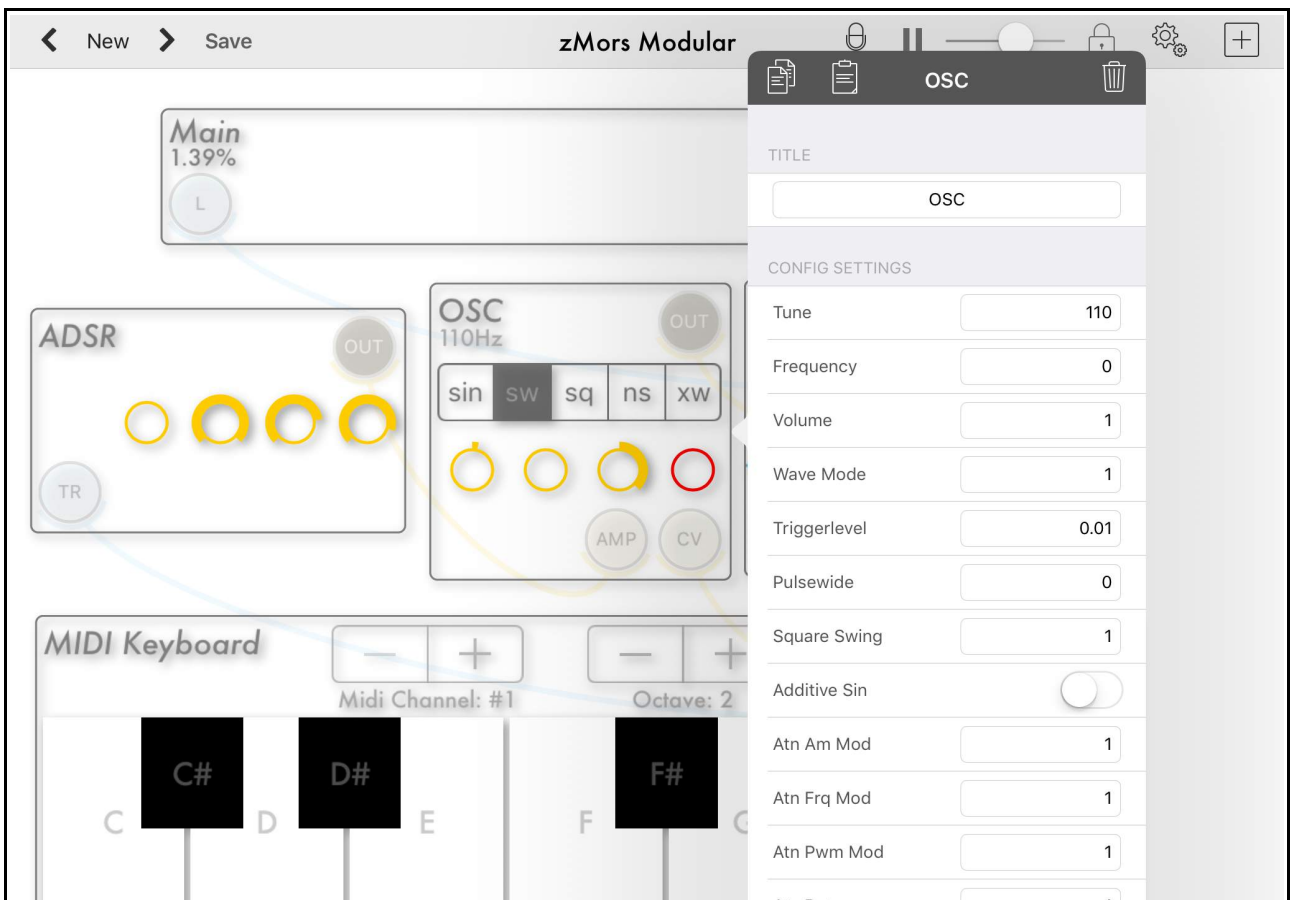


**Tap** on the **lock** icon to lock (hide) the cables.

The lock icon will appear red when locked.



To access the settings menu for any module, simply **tap and hold** on the **modules title**.

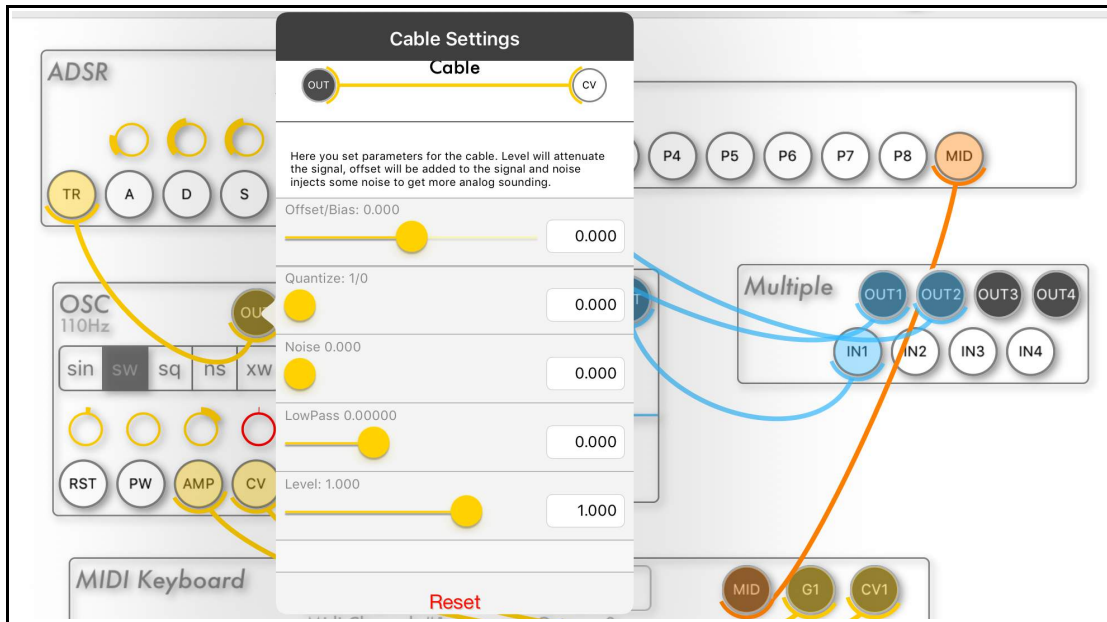


## Cable Settings

Tap on the **OUT** cable connection circle to access the cable setting menu.

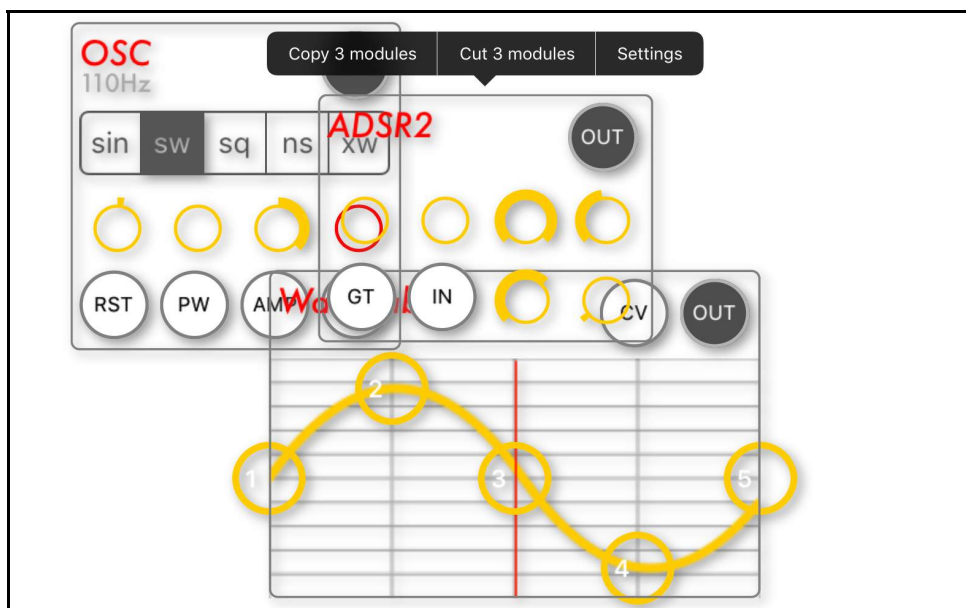


The cable modifiers include a low pass filter, noise, offset, quantize, level and reset.



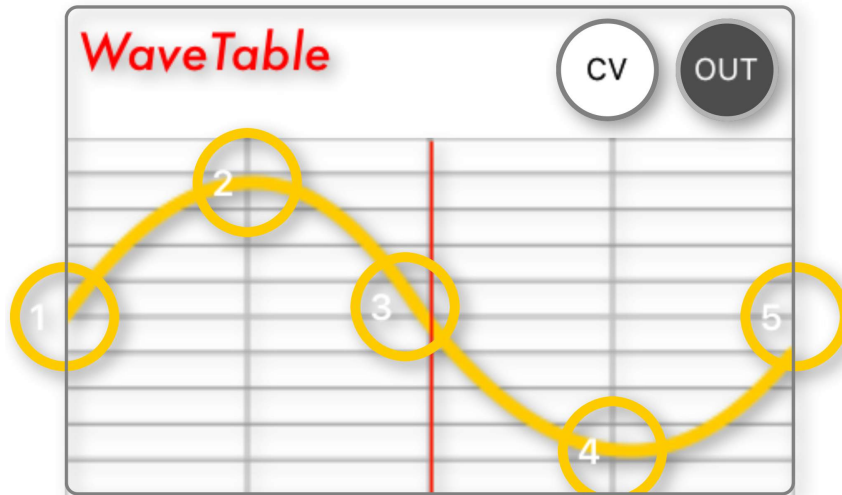
## The Grouping Modules Function

You may cut or copy multiple modules by dragging them together so they overlap. Then tap & hold to bring up the option bar.



# Exploring The WaveTable & Sampler Modules

## Using The WaveTable Generator



Tap and hold on the Generators title to access the settings menu.

**WaveTable**

TITLE  
WaveTable

CONFIG SETTINGS

Preset >

Additive

Unipolar

Osc Mode

Tune 440

PORT NAME - MIDI CHANNEL/CC

CV Ch 1-16 CC 0-127

OUT Ch CC

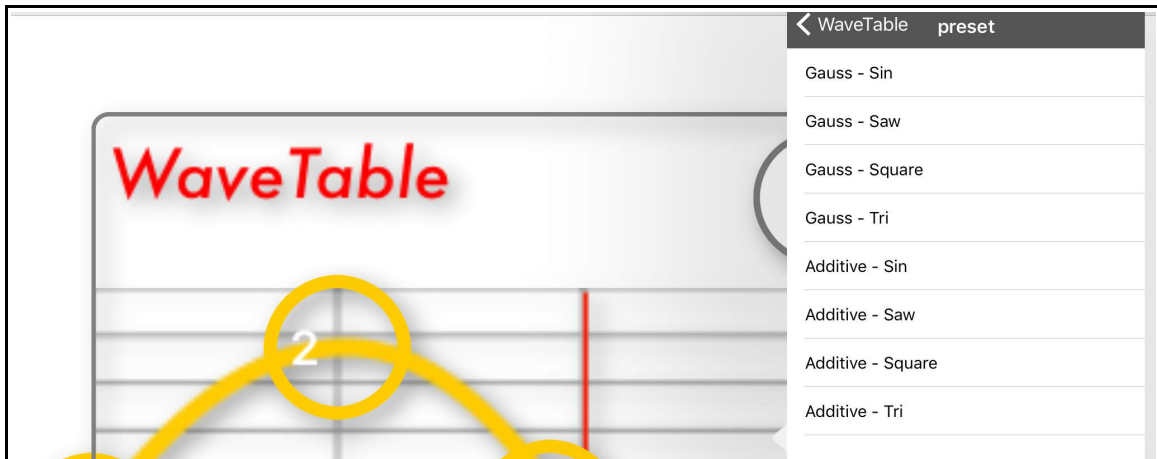
HELP

This is a wavetable oscillator or lookup table. An input value of range -1/+1 will be translated into a new value defined by the curve. Tap and hold in a head to select preset wave forms. The algorithm is based on gaussian elimination or additive sin gen. Values higher than 1.0 will be clipped. In oscillator mode (default) you can define tune in context menu and use input as cv pitch select. If

Tap on the **Preset** option in the menu to access the available waveforms (see below).



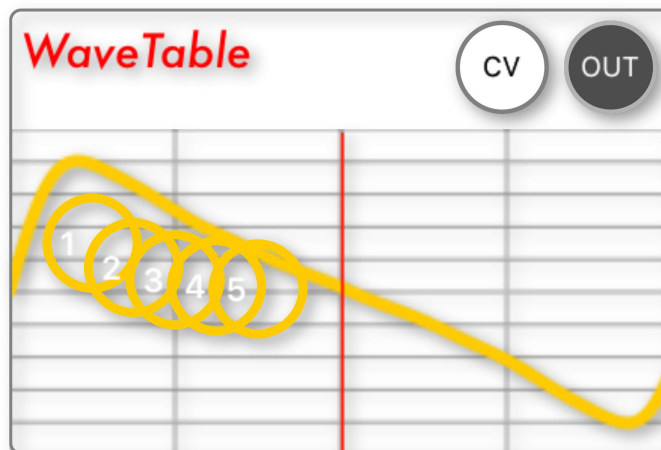
Below is the preset waveform menu for the WaveTable.



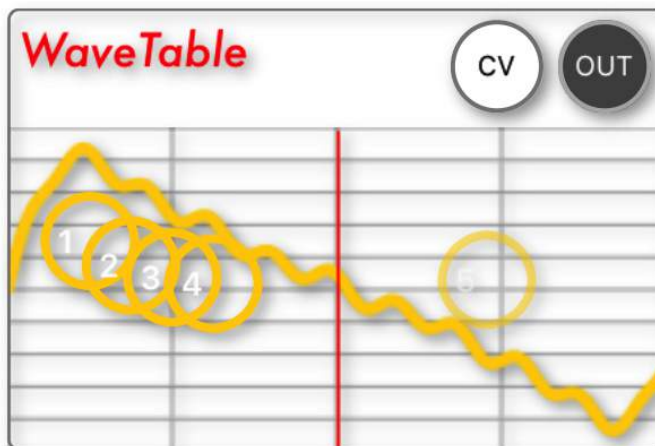
The wavetable has 2 modes: **gauss** and **additive**.

Classic waveforms can be used for each of these modes. Preset CV input can act as pitch control or phase lookup, depending on the mode used (the default is pitch oscillator).

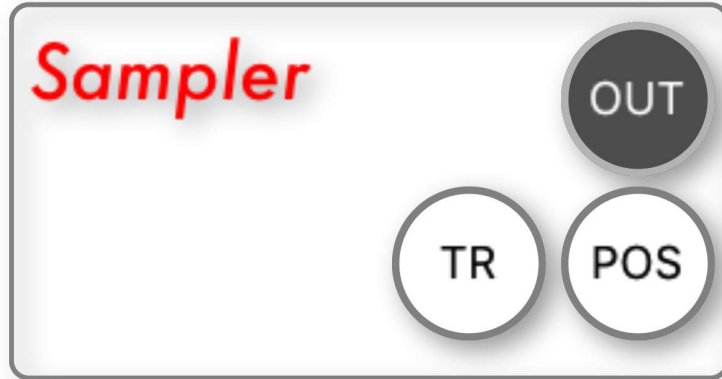
As an example, select the Gauss – Saw option.



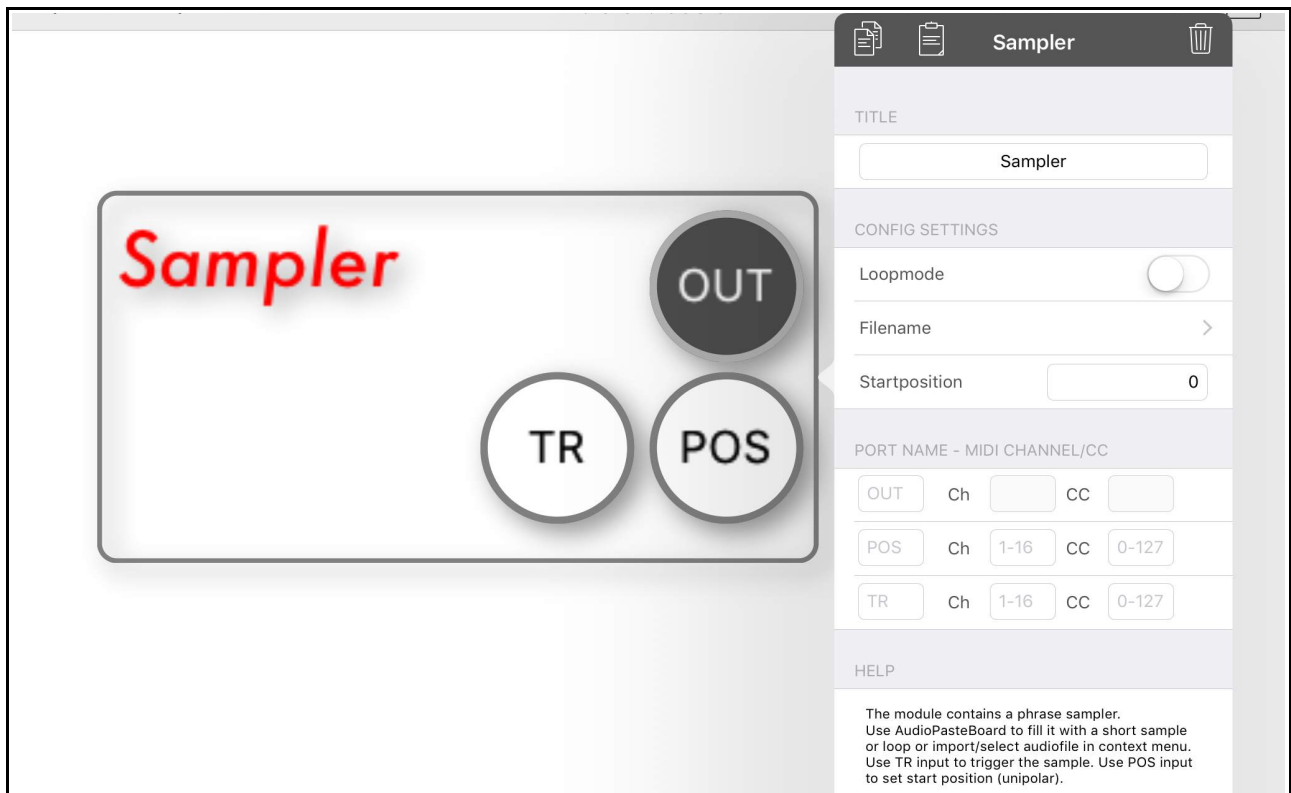
You can further shape the waveform using the **tap and drag** method with any of the numbered circles. See the waveform change in real time as you move your finger.



## Using The Sampler Module



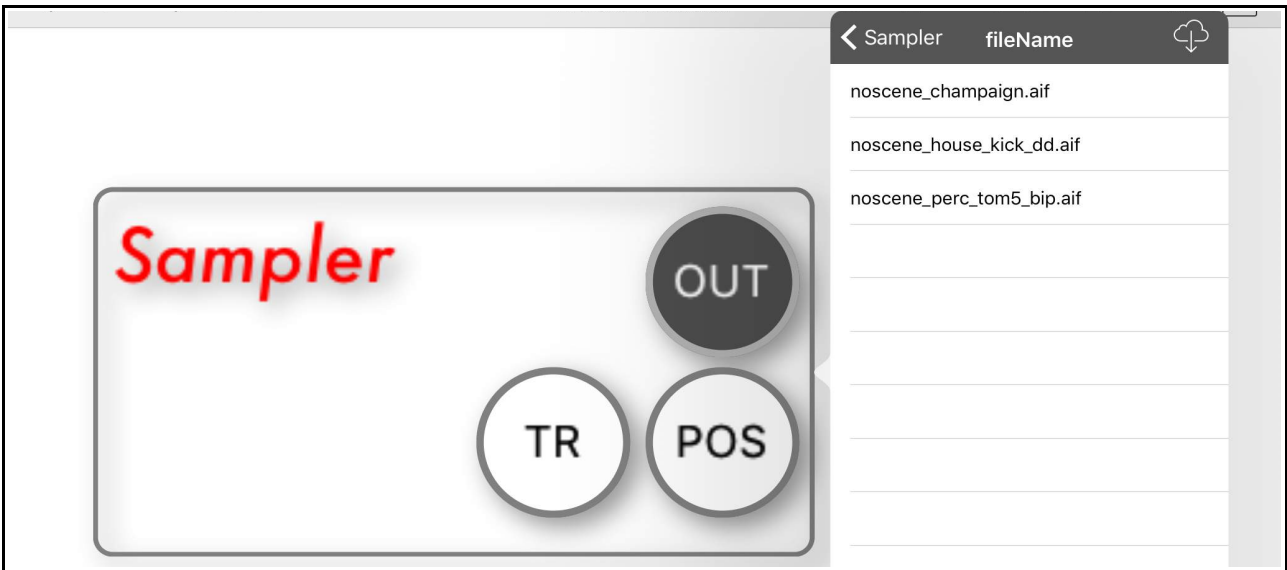
Use the **tap & hold** method on the modules title to access the settings menu.



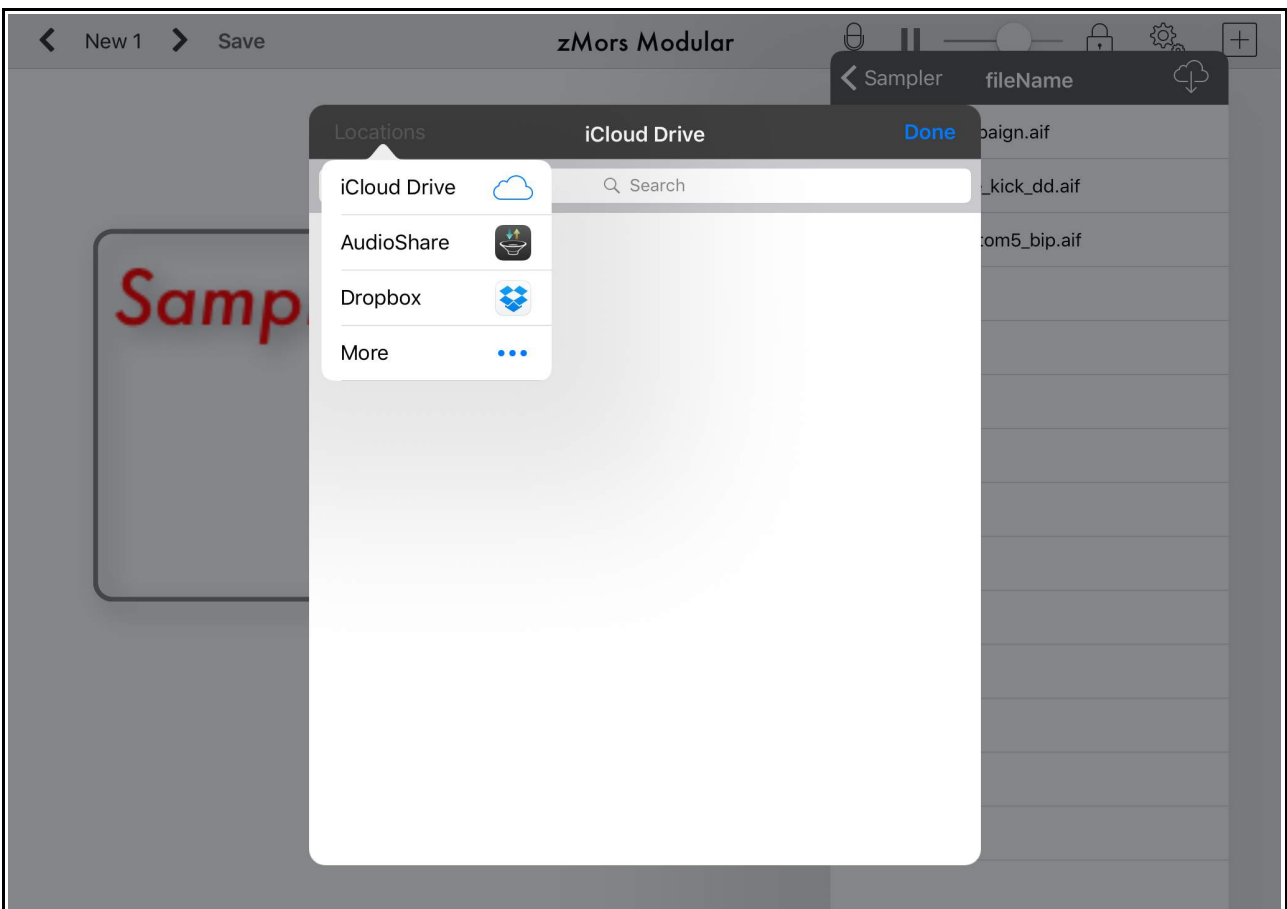
**Tap** on *Filename* in the Sampler menu to access the samples (See below).



The **Filename menu** allows you to load your samples into the sampler module.

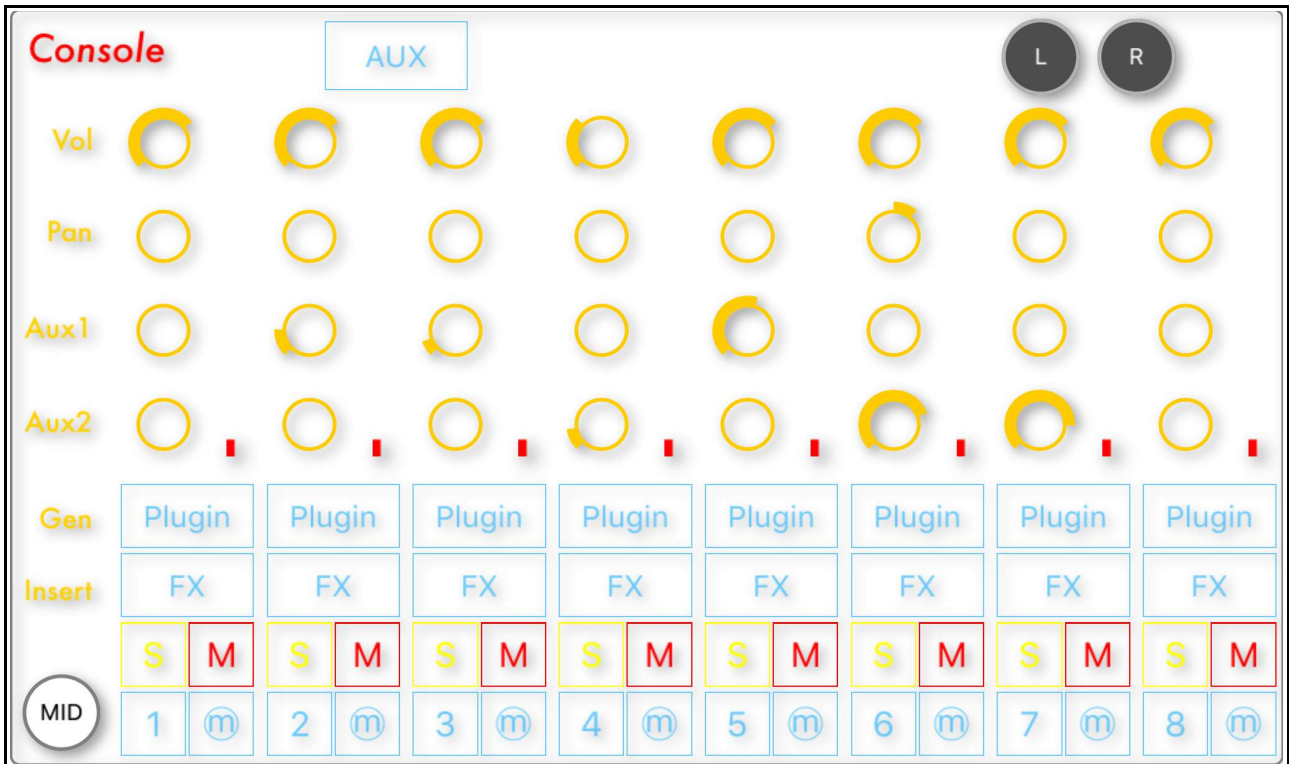


Tap on the **cloud** icon to access the file sharing options to import and/or export samples.



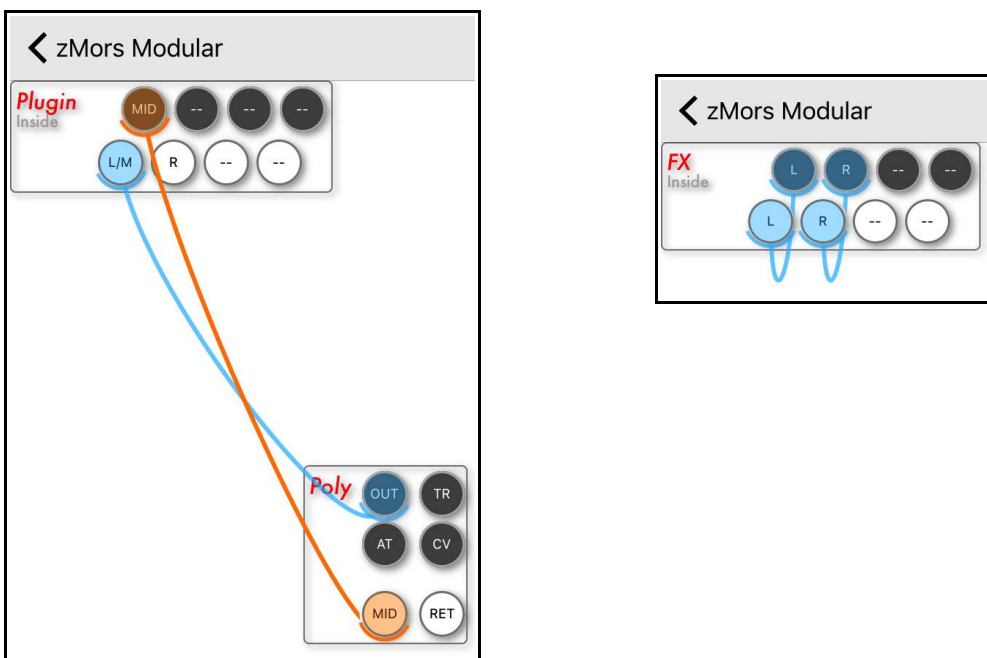
# THE GUI Options

## Using The Console Module

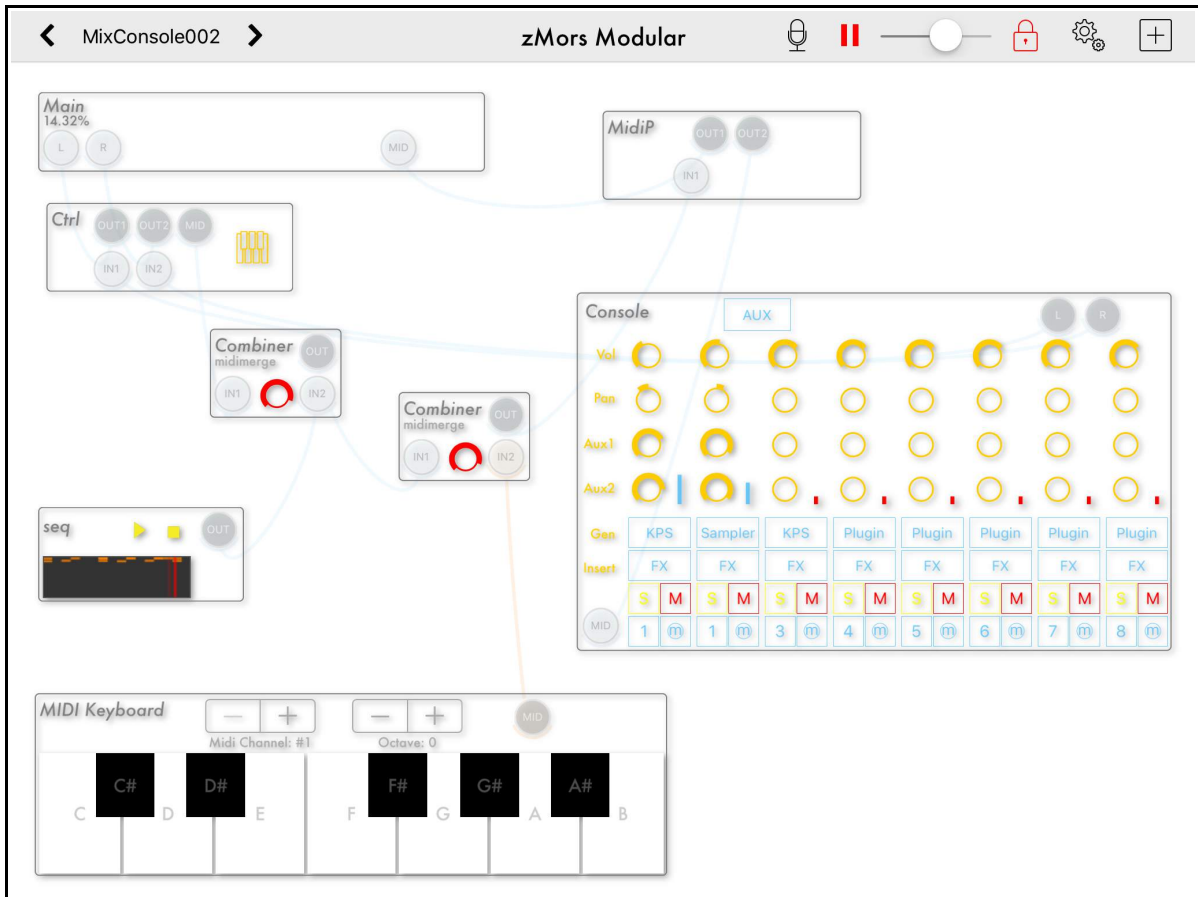


The Console Module is a preconfigured mixing console, ready to be connected into further modules using MIDI and audio in connectors.

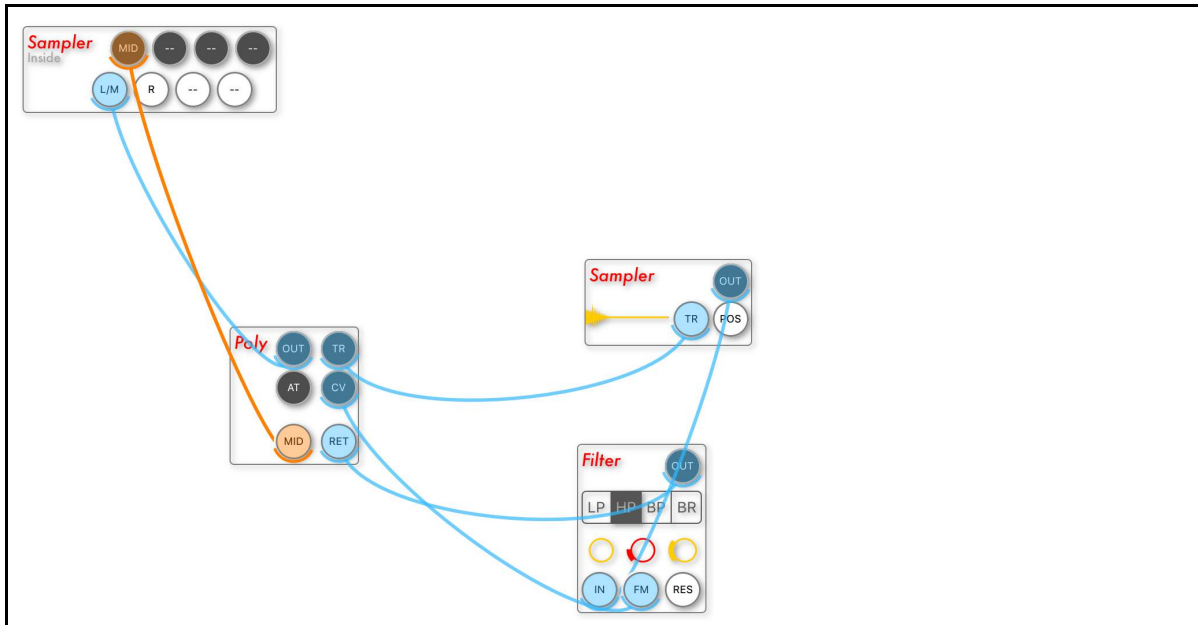
**Tap** on **Plugin** or **FX** to access the modules associated with each.



Below is an example from the Preset List of the **Console** in use -



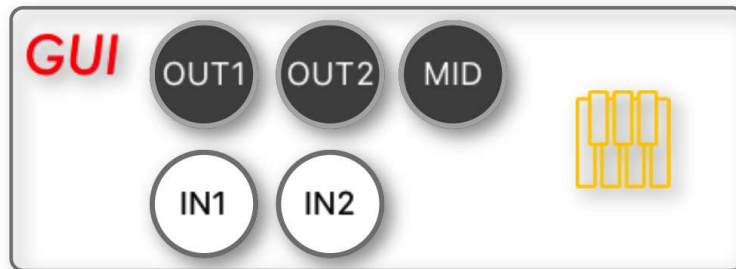
**Tap** on the **Sampler** button (Plugin) to see the full patch.



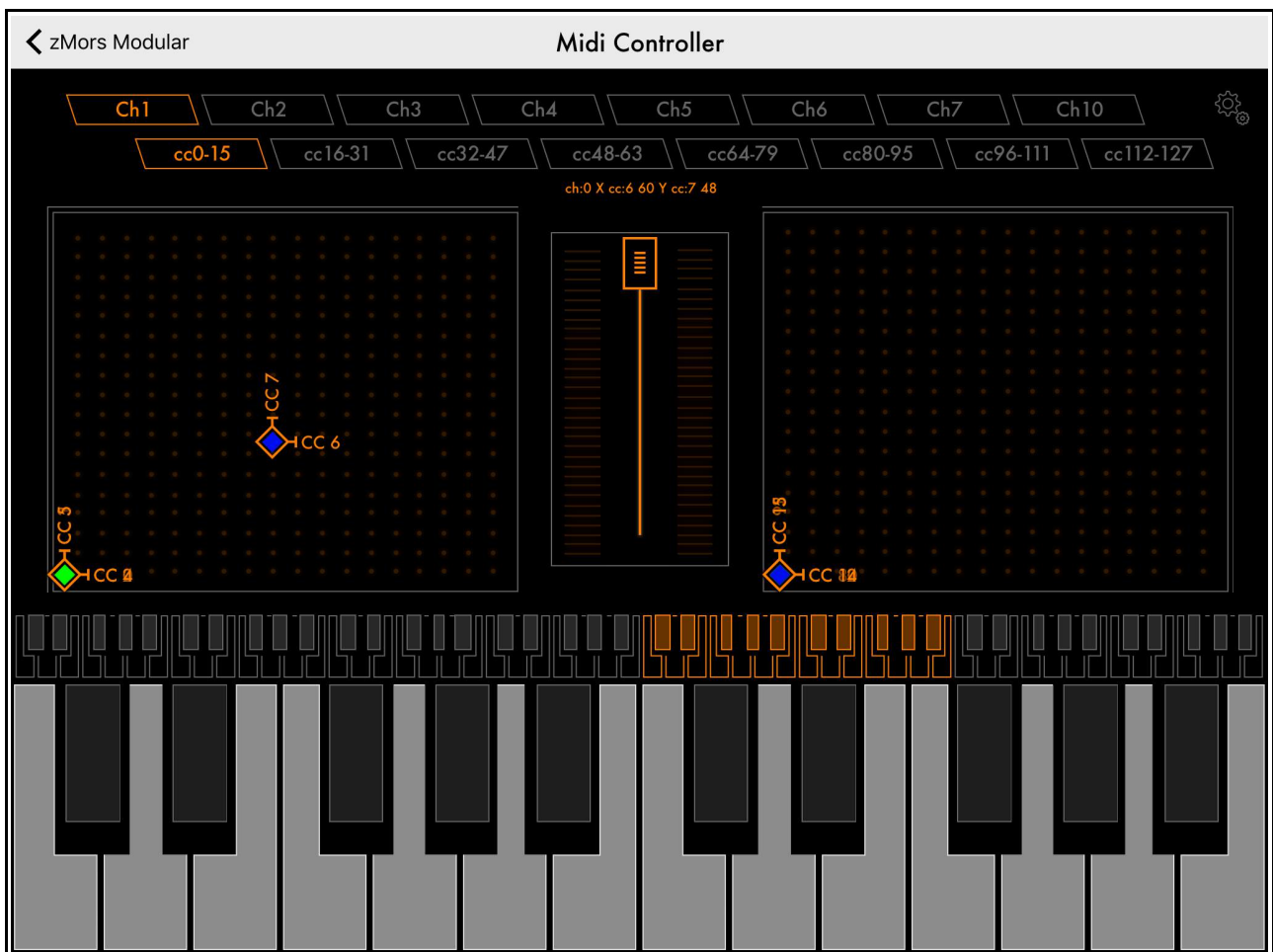
**Please note** - Don't forget to set the polyphony in the patch above.

To do this - **Tap and hold** on the **Poly** module to access it's settings menu, then use the **Max Poly** setting to specify the maximum polyphony you require.

## Using The JSCtrl Module

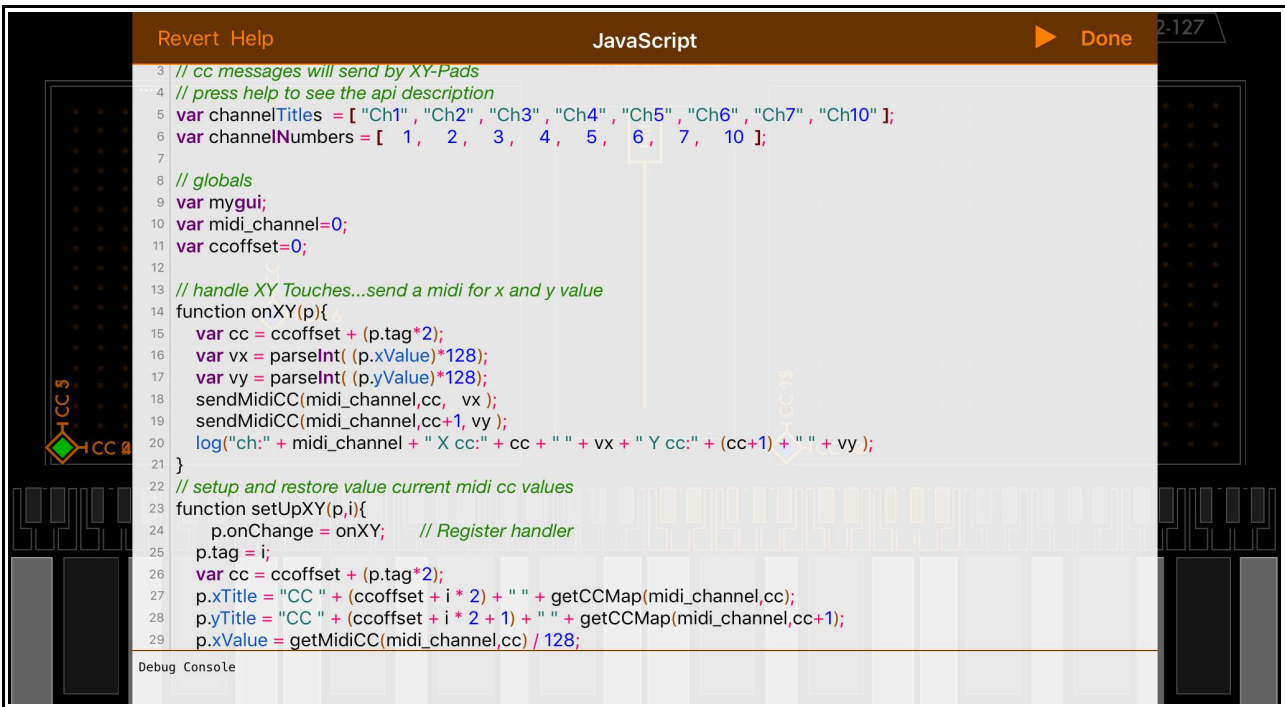


**Tap** on the **yellow keys** icon to access the **MIDI Controller** interface.



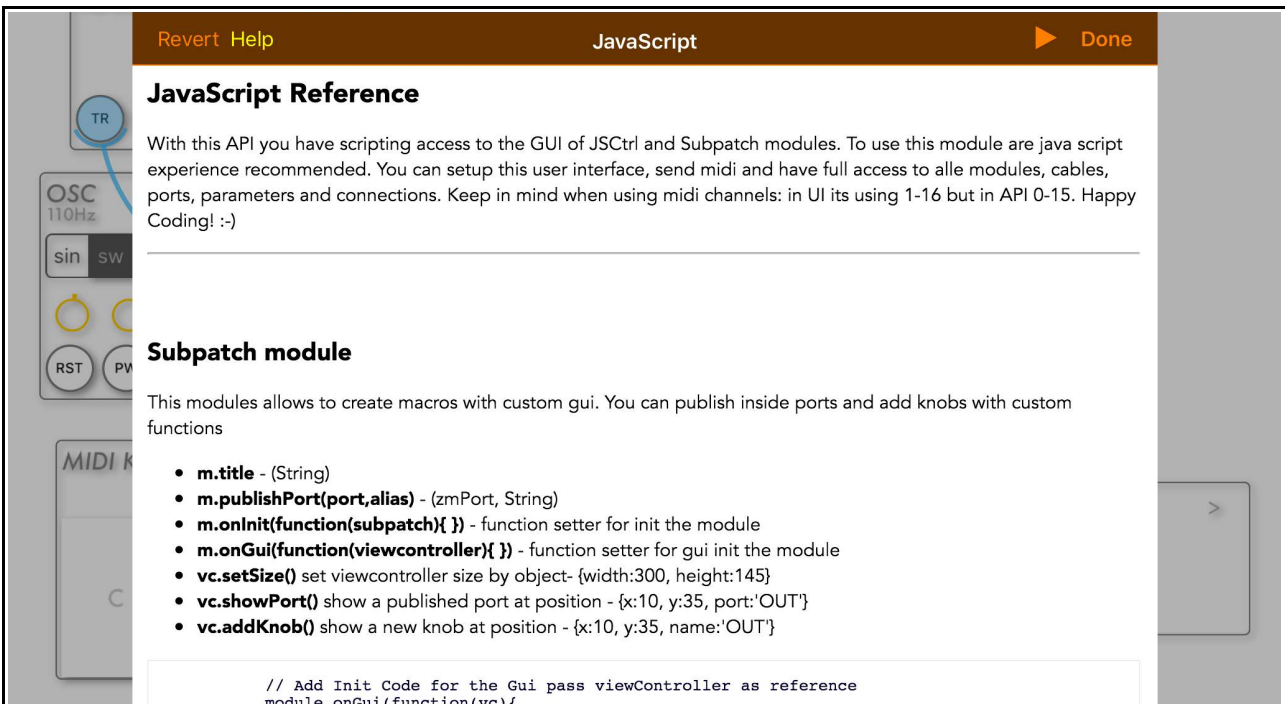
You can access the **JavaScript** code for the MIDI Controller using the **gear** icon (see below).

Here you can add to the code or create your own script.



```
3 // cc messages will send by XY-Pads
4 // press help to see the api description
5 var channelTitles = [ "Ch1", "Ch2", "Ch3", "Ch4", "Ch5", "Ch6", "Ch7", "Ch10" ];
6 var channelNumbers = [ 1, 2, 3, 4, 5, 6, 7, 10 ];
7
8 // globals
9 var mygui;
10 var midi_channel=0;
11 var ccoffset=0;
12
13 // handle XY Touches...send a midi for x and y value
14 function onXY(p){
15     var cc = ccoffset + (p.tag*2);
16     var vx = parseInt( (p.xValue)*128);
17     var vy = parseInt( (p.yValue)*128);
18     sendMidiCC(midi_channel,cc, vx );
19     sendMidiCC(midi_channel,cc+1, vy );
20     log("ch:" + midi_channel + " X cc:" + cc + " + vx + " Y cc:" + (cc+1) + " + vy );
21 }
22 // setup and restore value current midi cc values
23 function setUpXY(p,i){
24     p.onChange = onXY; // Register handler
25     p.tag = i;
26     var cc = ccoffset + (p.tag*2);
27     p.xTitle = "CC " + (ccoffset + i * 2) + " " + getCCMap(midi_channel,cc);
28     p.yTitle = "CC " + (ccoffset + i * 2 + 1) + " " + getCCMap(midi_channel,cc+1);
29     p.xValue = getMidiCC(midi_channel,cc) / 128;
```

**Tap on Help** to learn more about using **JavaScript** to design your own interface.



### JavaScript Reference

With this API you have scripting access to the GUI of JSCtrl and Subpatch modules. To use this module are java script experience recommended. You can setup this user interface, send midi and have full access to alle modules, cables, ports, parameters and connections. Keep in mind when using midi channels: in UI its using 1-16 but in API 0-15. Happy Coding! :-)

#### Subpatch module

This modules allows to create macros with custom gui. You can publish inside ports and add knobs with custom functions

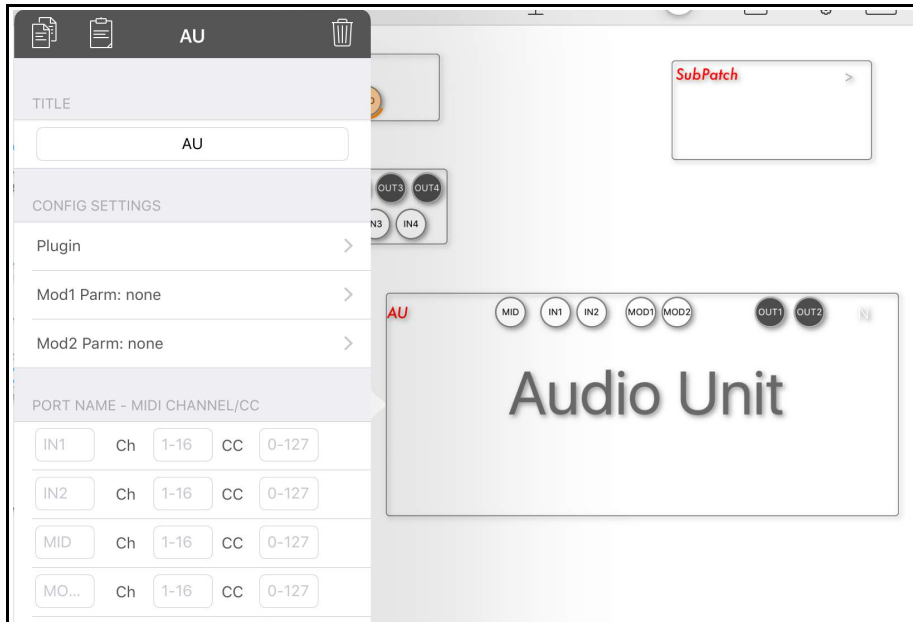
- **m.title** - (String)
- **m.publishPort(port,alias)** - (zmPort, String)
- **m.onInit(function(subpatch){ })** - function setter for init the module
- **m.onGui(function(viewcontroller){ })** - function setter for gui init the module
- **vc.setSize()** set viewcontroller size by object- {width:300, height:145}
- **vc.showPort()** show a published port at position - {x:10, y:35, port:'OUT'}
- **vc.addKnob()** show a new knob at position - {x:10, y:35, name:'OUT'}

```
// Add Init Code for the Gui pass viewController as reference
module.onGui(function(vc){
```

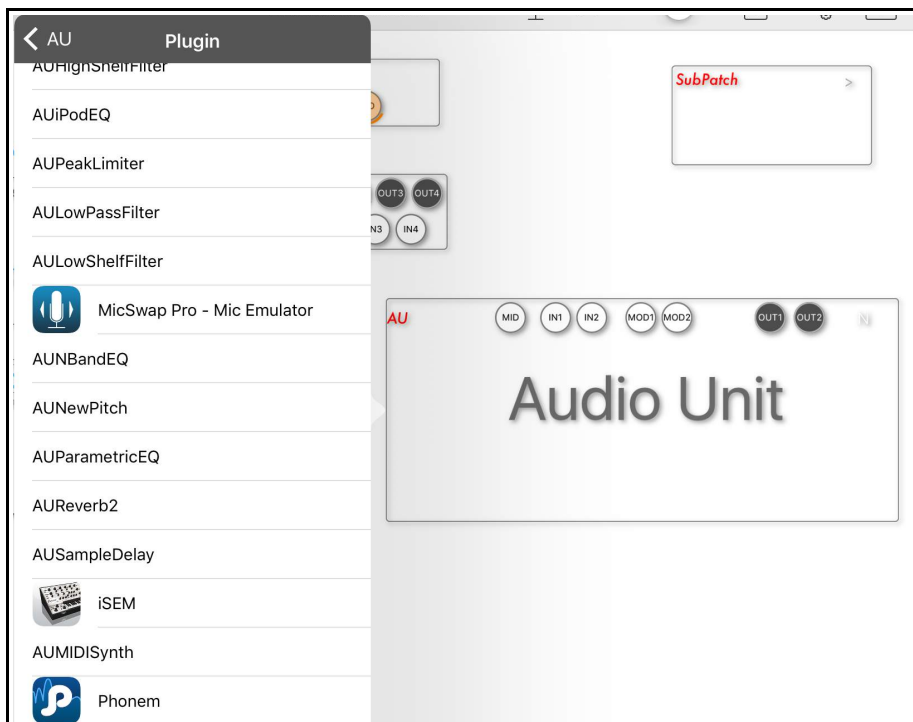
# Exploring The AU & IAA Modules

## Using the Audi Unit (AU)

Use the **tap & hold** method to access the settings menu.



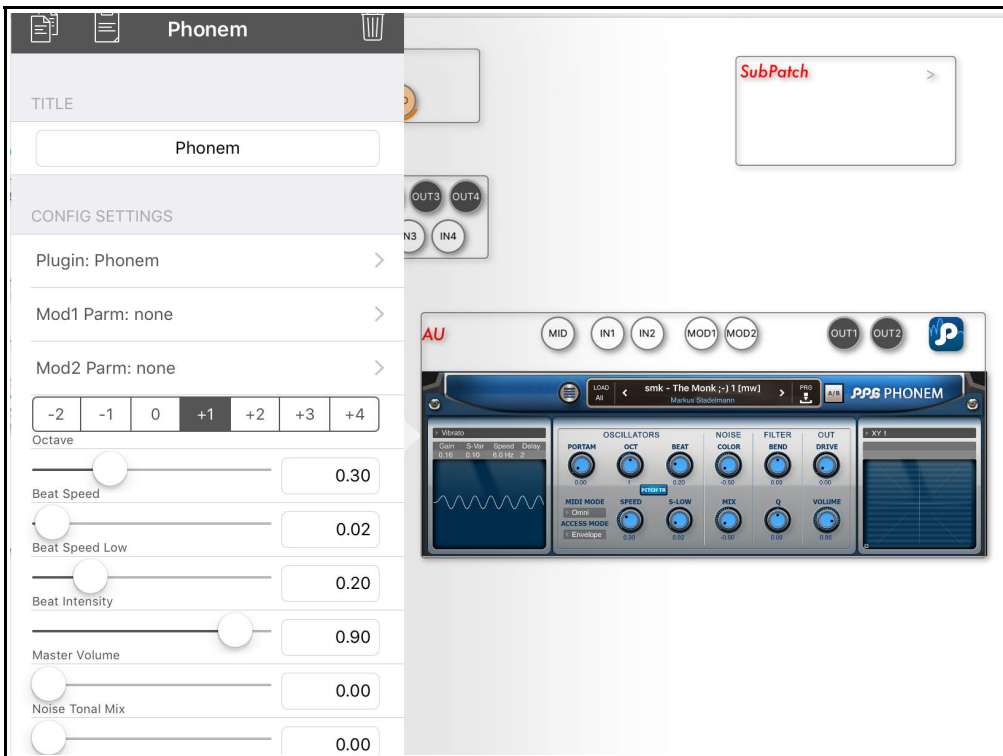
**Tap on Plugin** to select from the available **Audio Units**.



Select an **Audio Unit**.

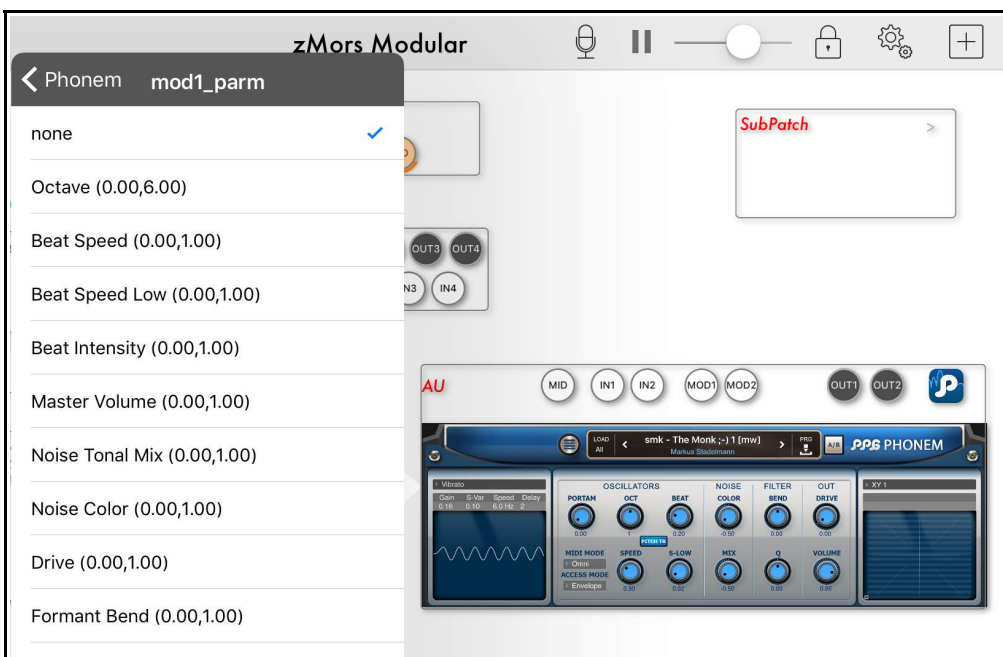


We will choose **Phonem** for this example (see below).  
 Now the AU settings menu will reflect the chosen Audio Unit.



**Tap on Mod1 Parm** to assign a modulation target.

The same process can be followed for the second modulation option (**Mod2 Parm**).



To be able to use the **AU interface**, you simply need to lock the preset.  
To do this, **tap** on the **lock** icon.

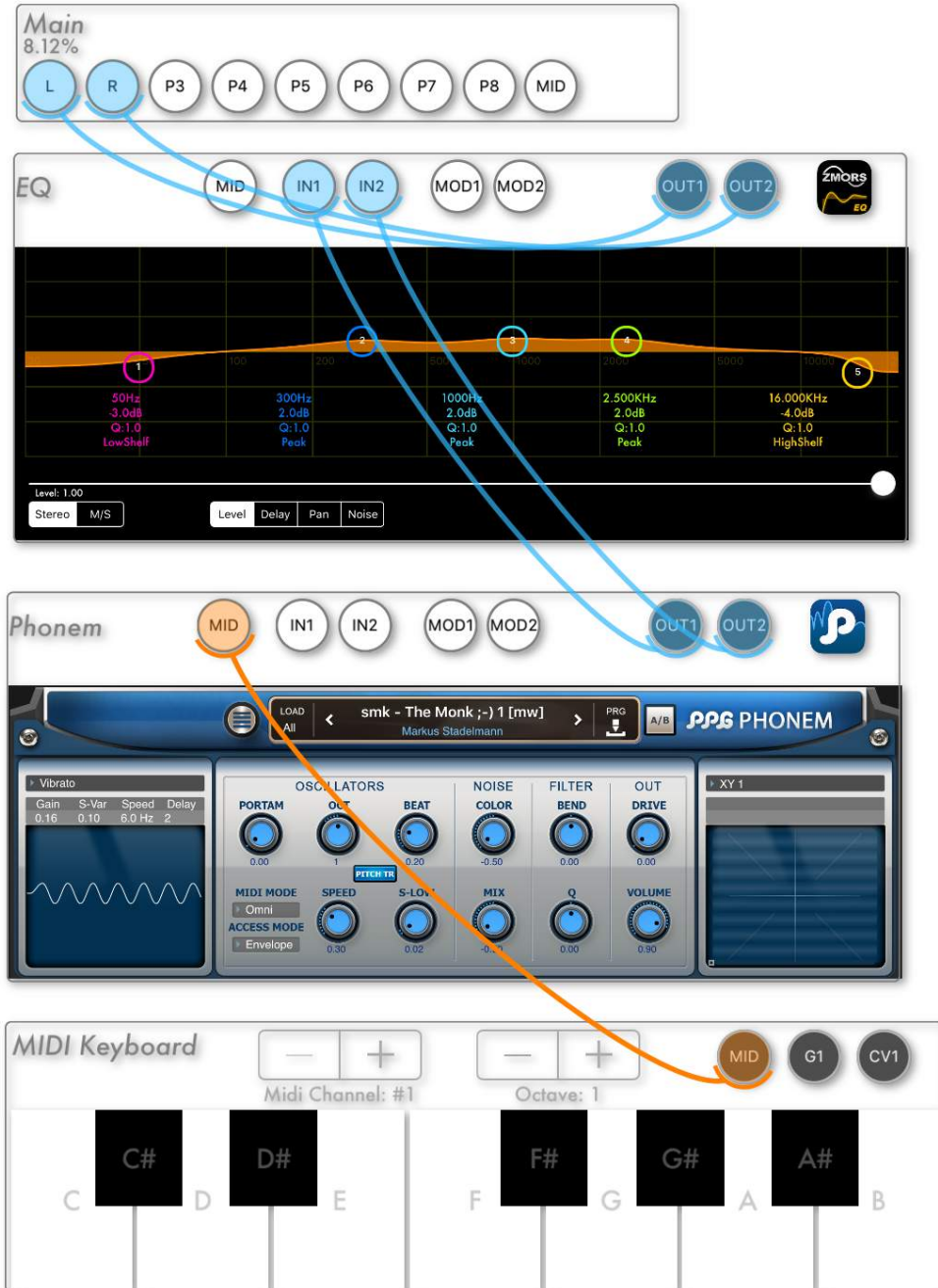


Now you can use the **AU** in real time, as part of your modular setup.

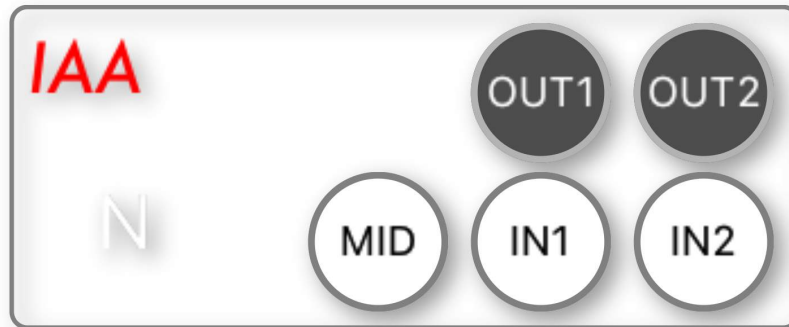
This means you can access the presets of the Audio Unit and manipulate all parameters.



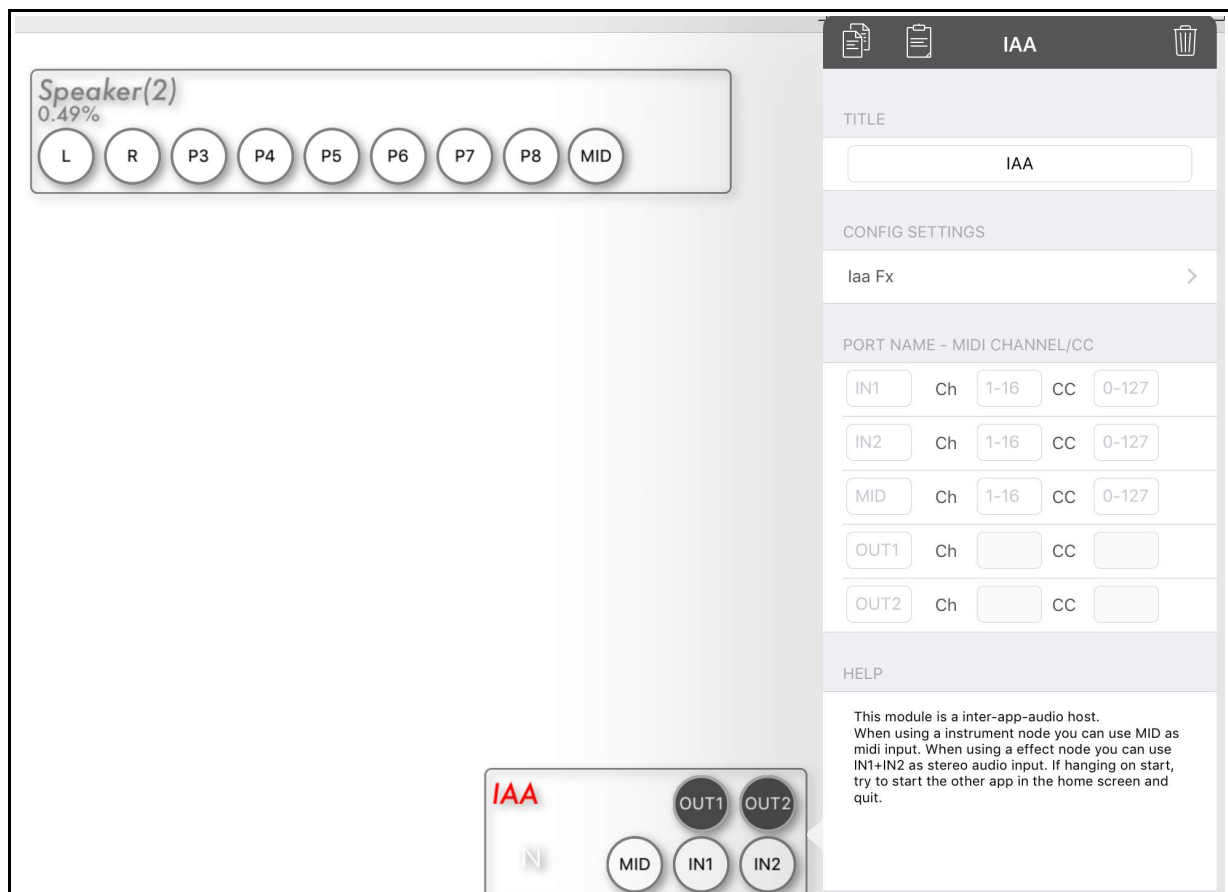
Below is an example of a simple AU v3 setup, playable from a keyboard.



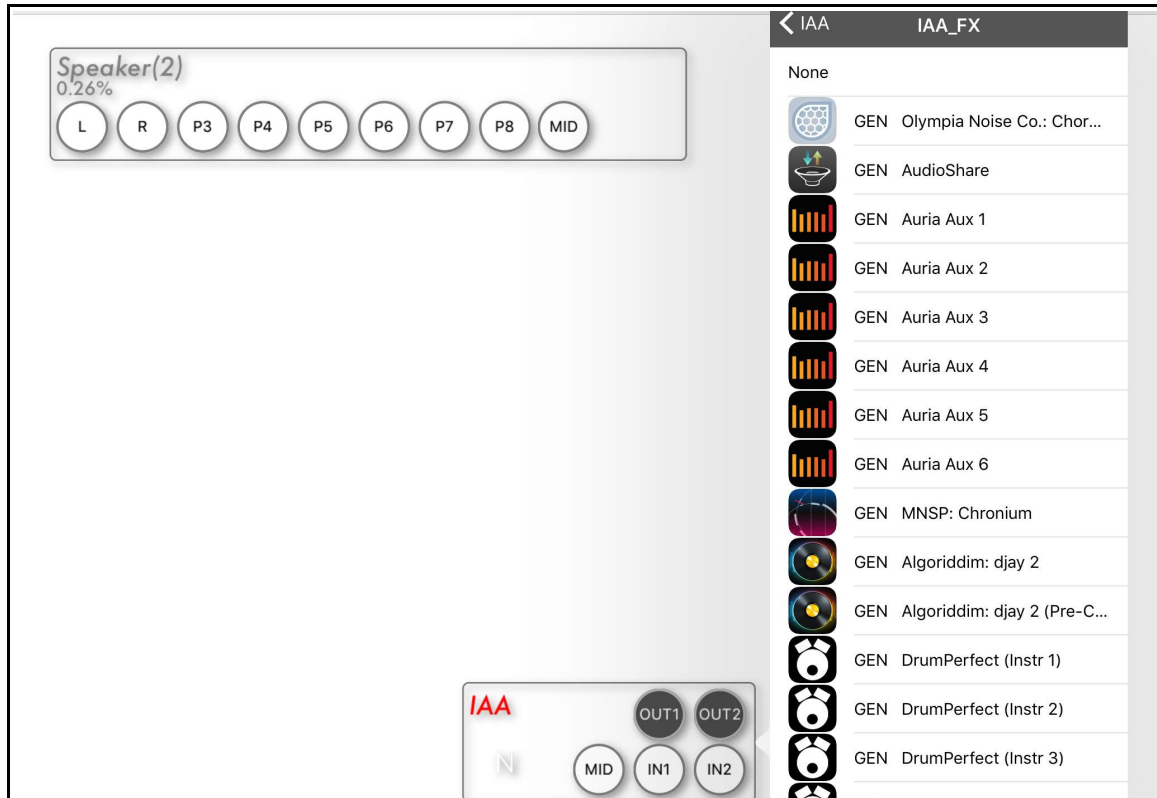
## Using The Inter App Audio (IAA)



**Tap & hold** on the **IAA module title** to access the settings menu.

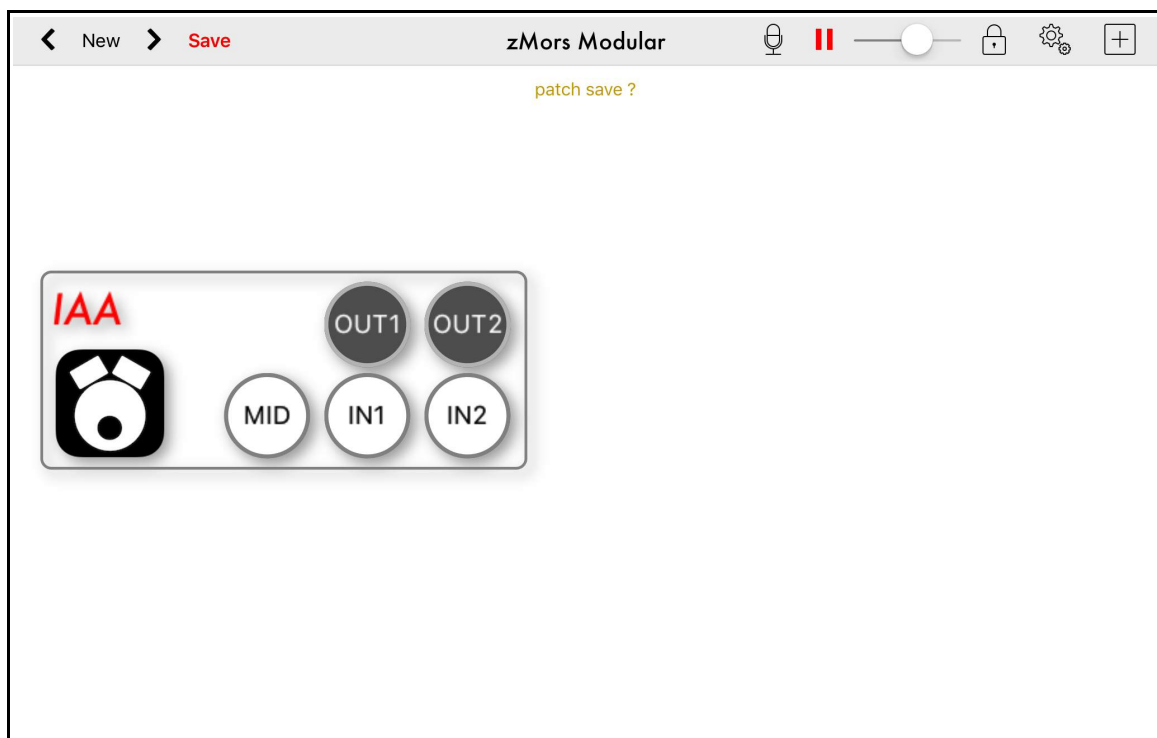


Tap on **iaa FX** to access the Inter App Audio options (see below).



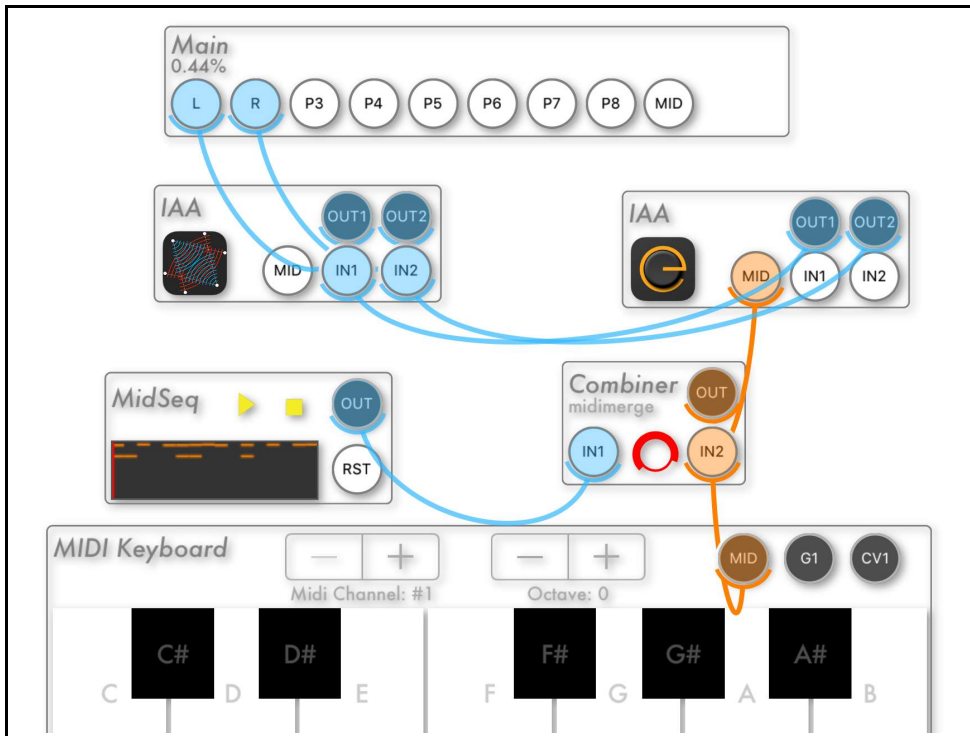
Select an IAA app or return to the previous menu.

For this example we will choose **GEN DrumPerfect (Instr 1)**.

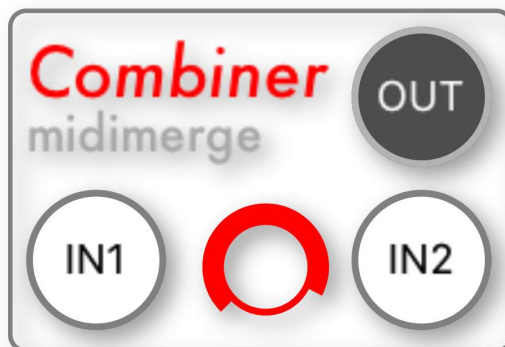
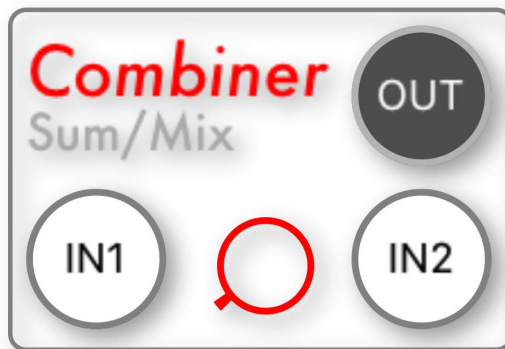


Tapping on the IAA app icon will take you directly to that app.

Below is an example of using the **Combiner Module** to mix the MIDI signals from the keyboard and the sequencer. Use the **red dial** on the Combiner to change the type to **midmerge**.

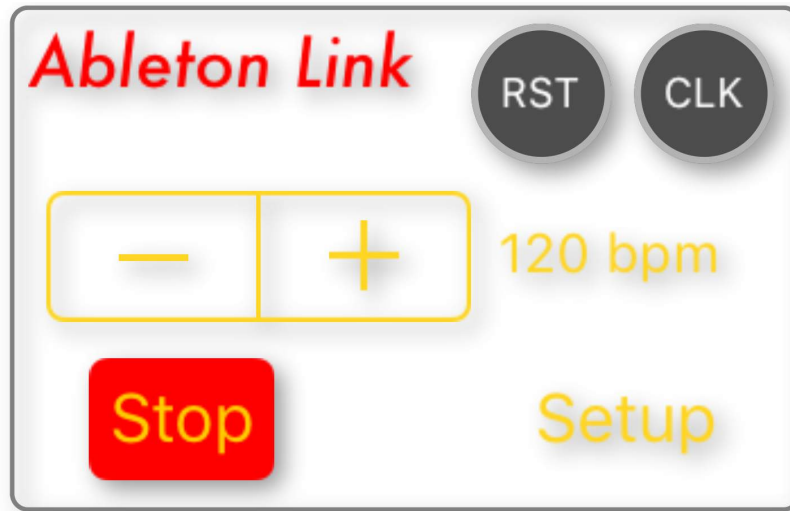


Two of the options for the combiner can be seen below.

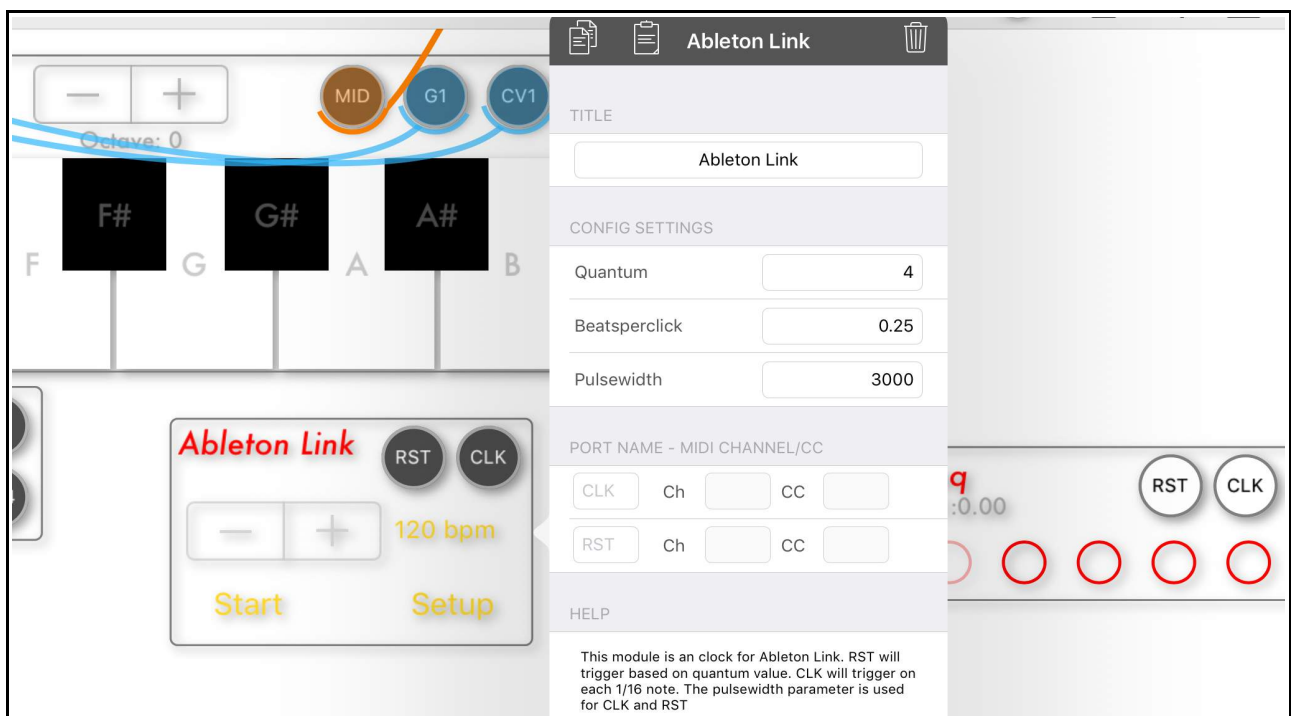


## Exploring The Seq & CV Modules

### Using the Ableton Link Clock Module



Use the **Tap & Hold** method on the Modules title to access the Settings Menu.



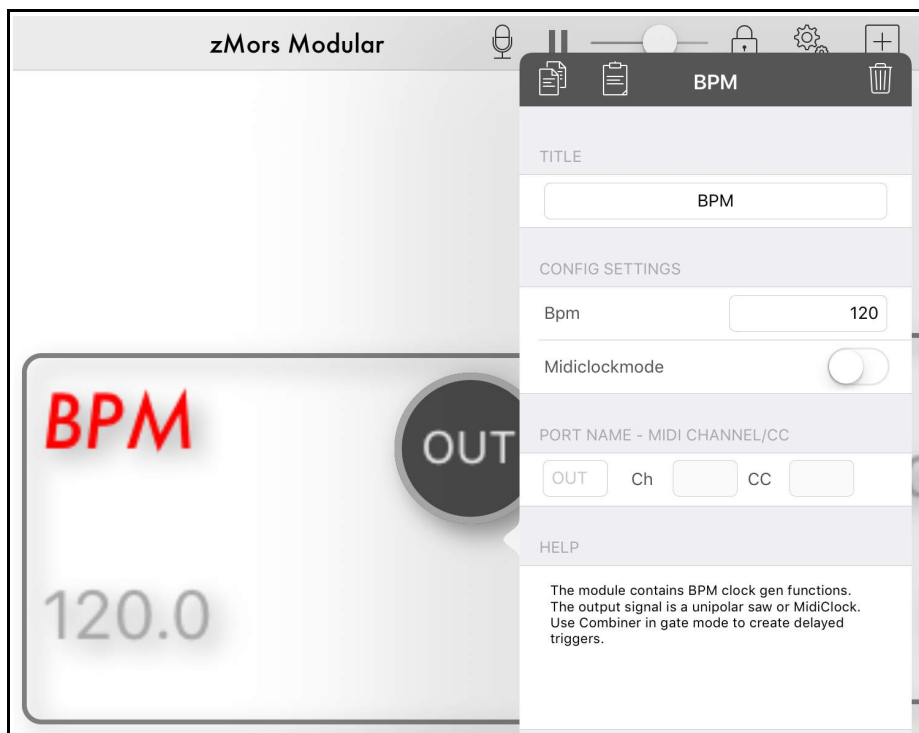
This is where you can set the **bar length** (Quantum) and the **clock division** (Beatsperclick).

Use the **Setup** button to turn the **Ableton Link** connection on.



## Using the BPM Module

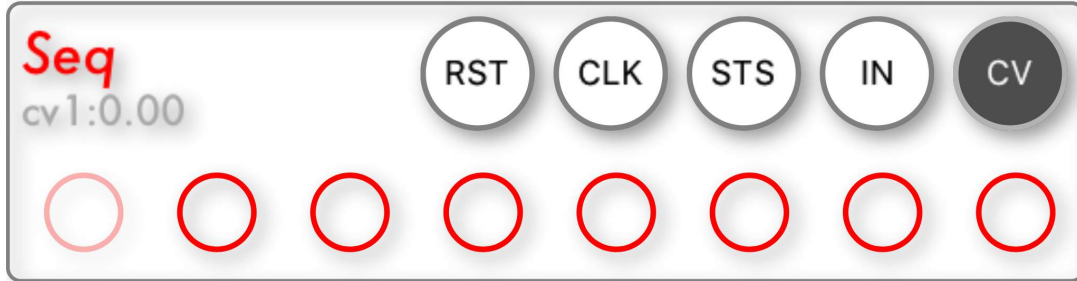
This module is used for setting the **BPM** and turning the **MIDI clock** on/off.



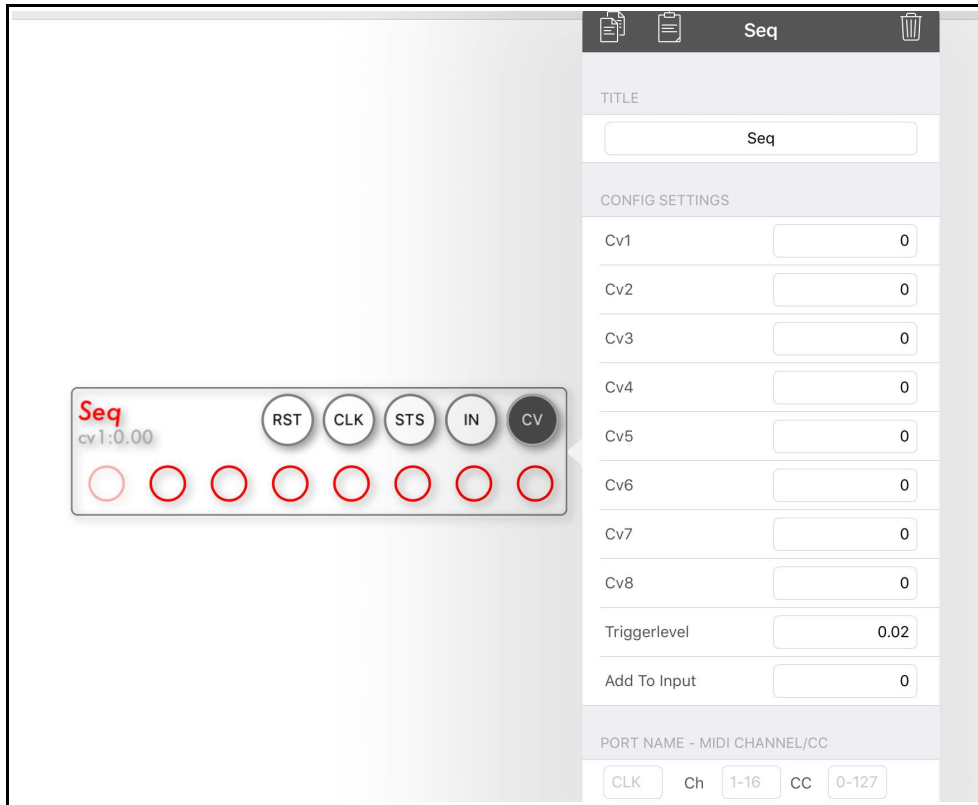


## Using the CV Sequencer Module

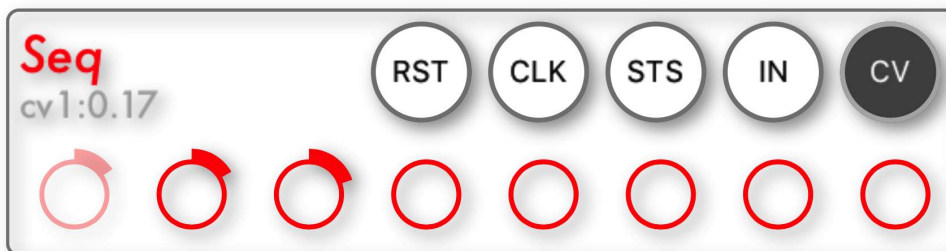
This is an 8 step analog-style Control Voltage sequencer to use with external hardware.



Use the **Tap & Hold** method to access the CV Sequencers Setting Menu.



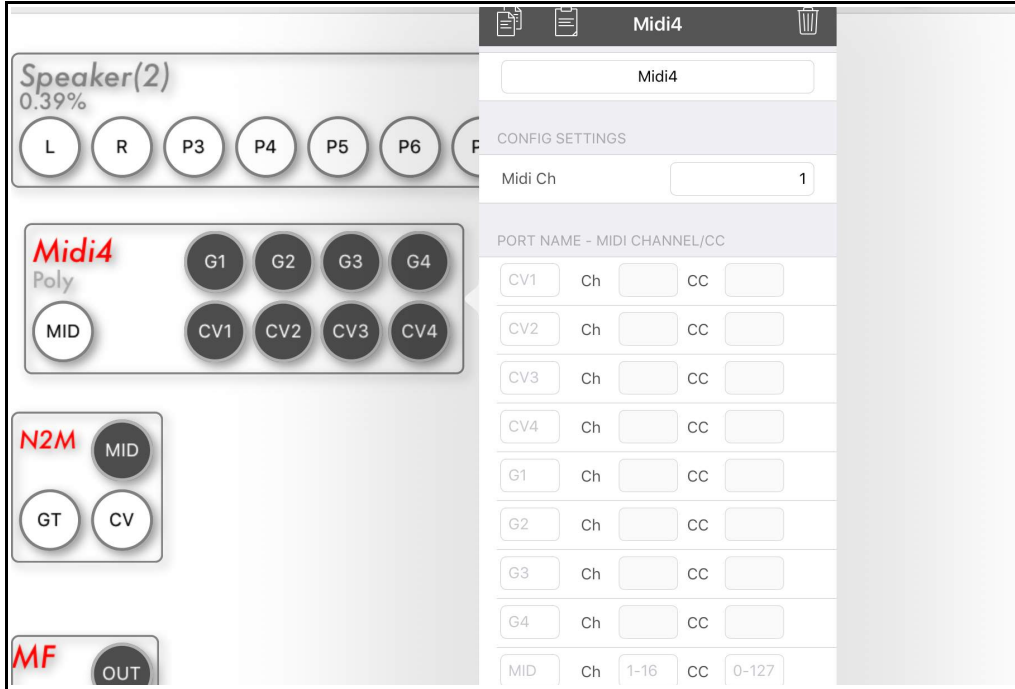
The CV amounts can be directly set using the red dials on the module.



**Please note** – The CV Sequencer can be used to shape waveforms by fast STS (step select) or CLK input (3 octaves lower).

## Using the Poly MIDI to CV/Gate (Midi4) Module

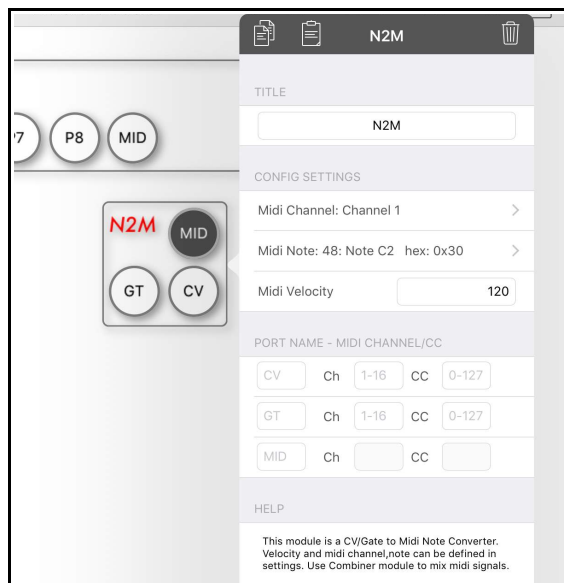
This is where you convert **polyphonic MIDI** signals to **CV** for use with **external hardware**. You can set the MIDI channel and MIDI to CV send channels.



## Using the CV/Gate to MIDI Converter (N2M) Module

This module converts **CV** signals back to **MIDI**. This allows you to accept signals back from **external hardware**.

You can select the MIDI channel and set the velocity in the settings menu.



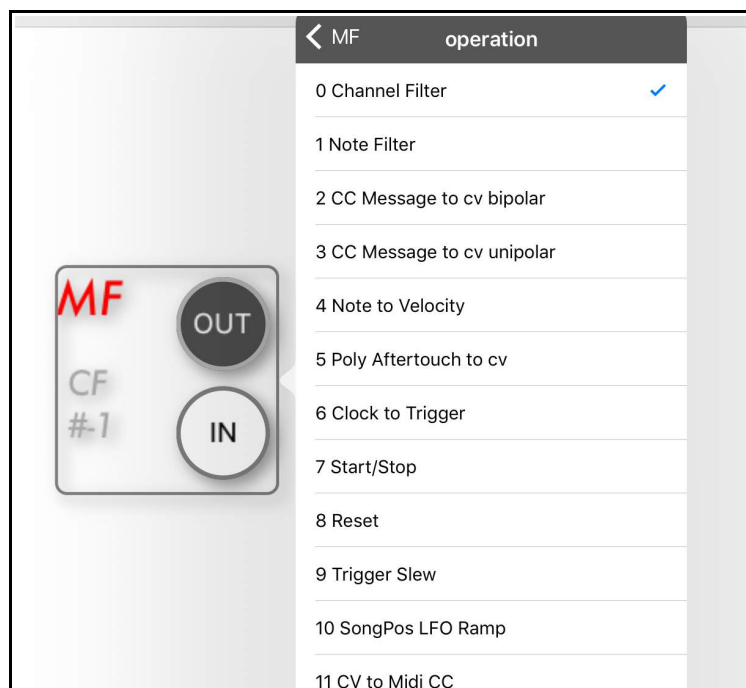
**Please note** - You need a DC coupled interface to produce note CV.

## Using the MIDI Filter Module

The MIDI Filter Module allows you to filter the MIDI information passing through.



**Tap on Operation** in the **settings menu** to access all the available **MIDI Filters**.



**Tap** on **Channel** to choose the MIDI channel.

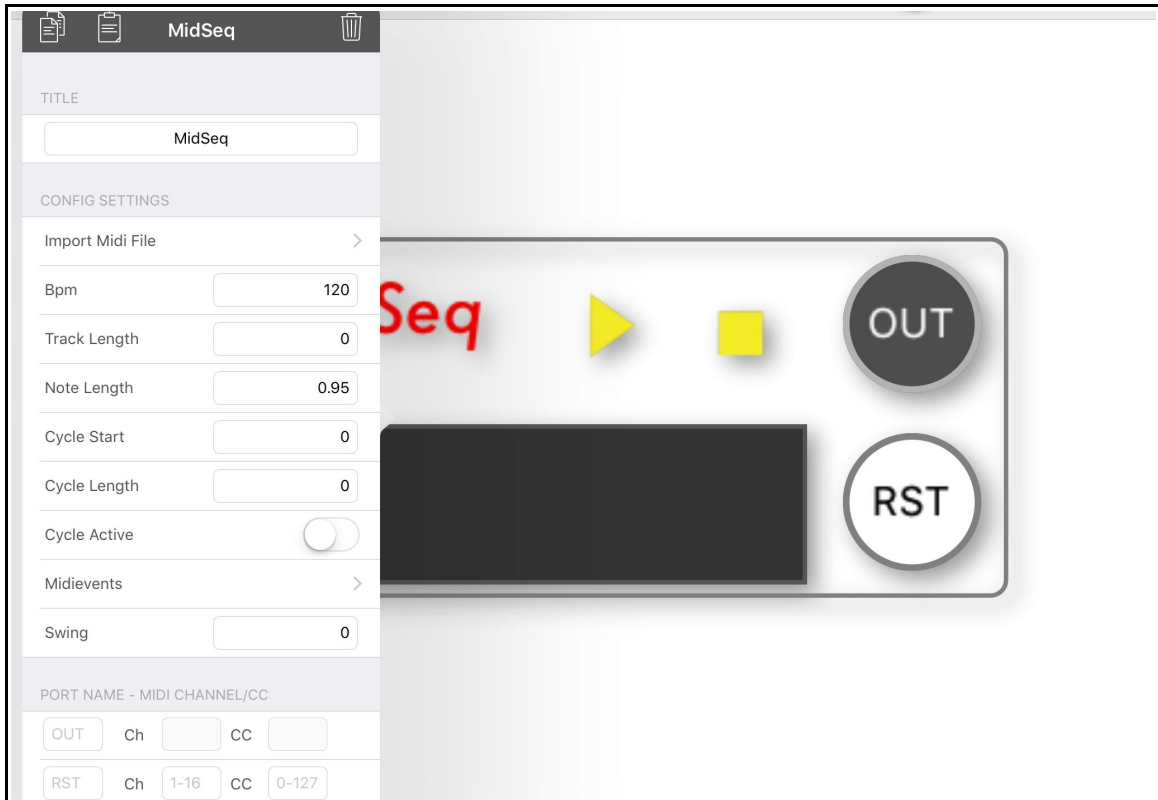
**Please note** – this module can be used to extract CLK and RST from the MIDIClock, and can convert CV to MIDI (& MIDI to CV).

## Using the MIDI Sequencer Module

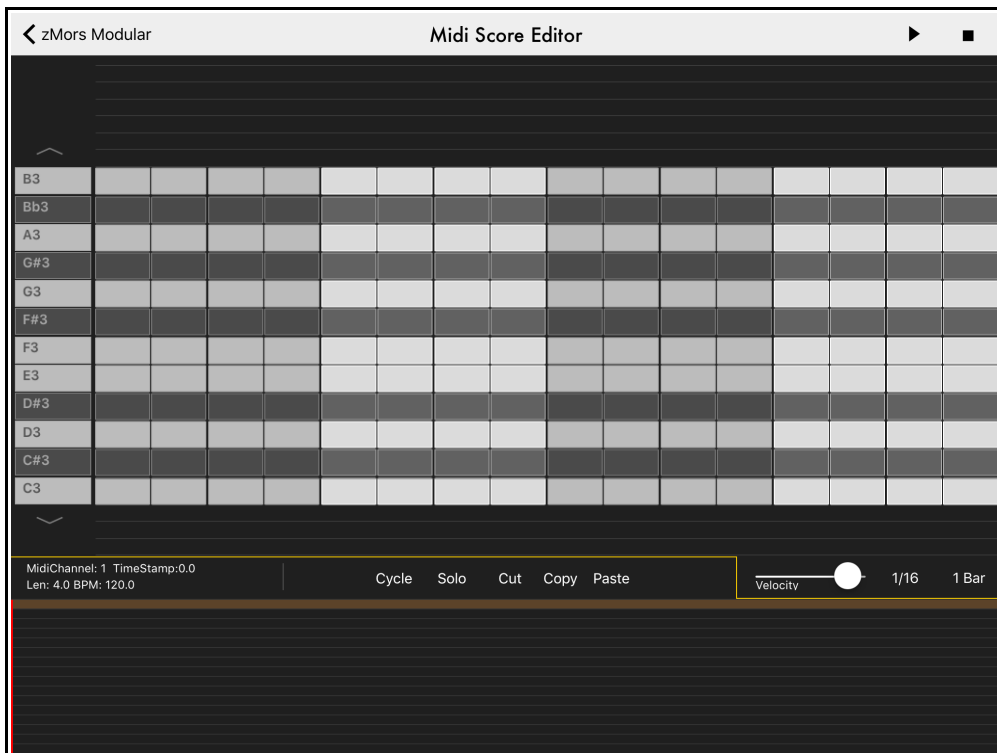


Use the **tap & hold** method to access the settings menu.

Here you can import MIDI files and adjust various parameters for the sequencer.

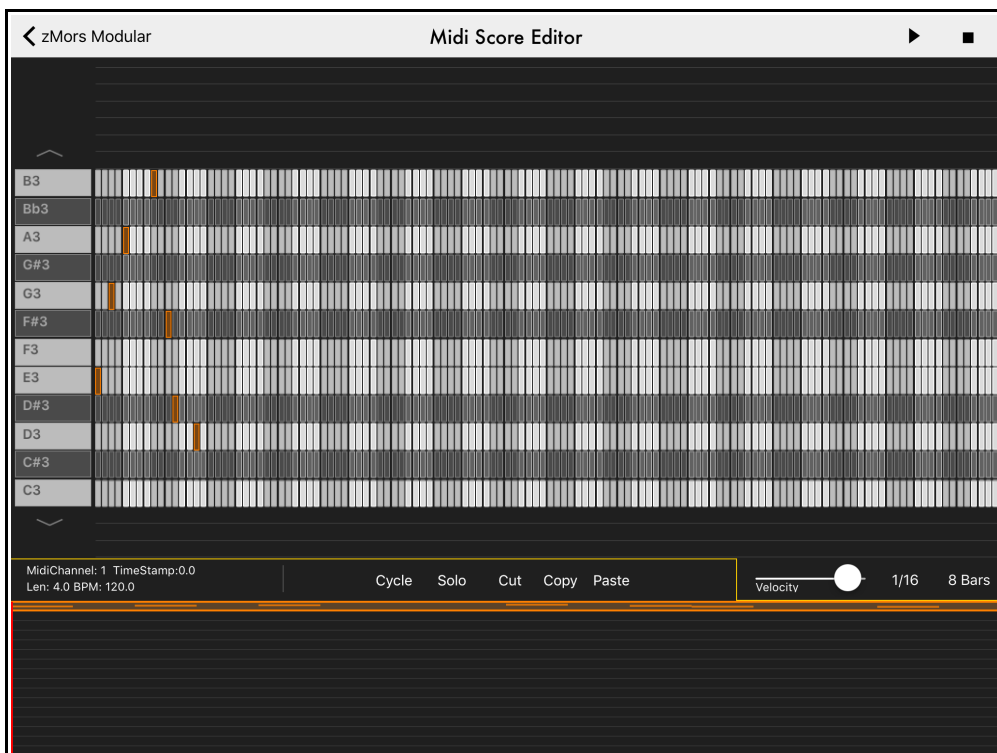


**Tap** on the black rectangle to access the piano roll style MIDI Score Editor (see below).



Here you find the *piano roll* style interface.  
Simple **tap** on the editor to add notes to the sequence.

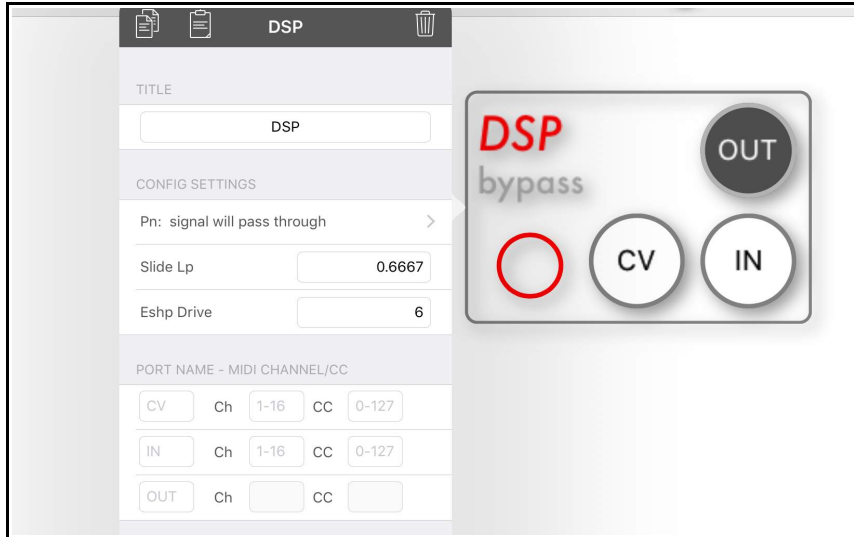
You can also extend the editor out to 8 bars, and use the **Cycle** option for looping an area.  
It is also possible to **cut**, **copy** and **paste** selections of notes.



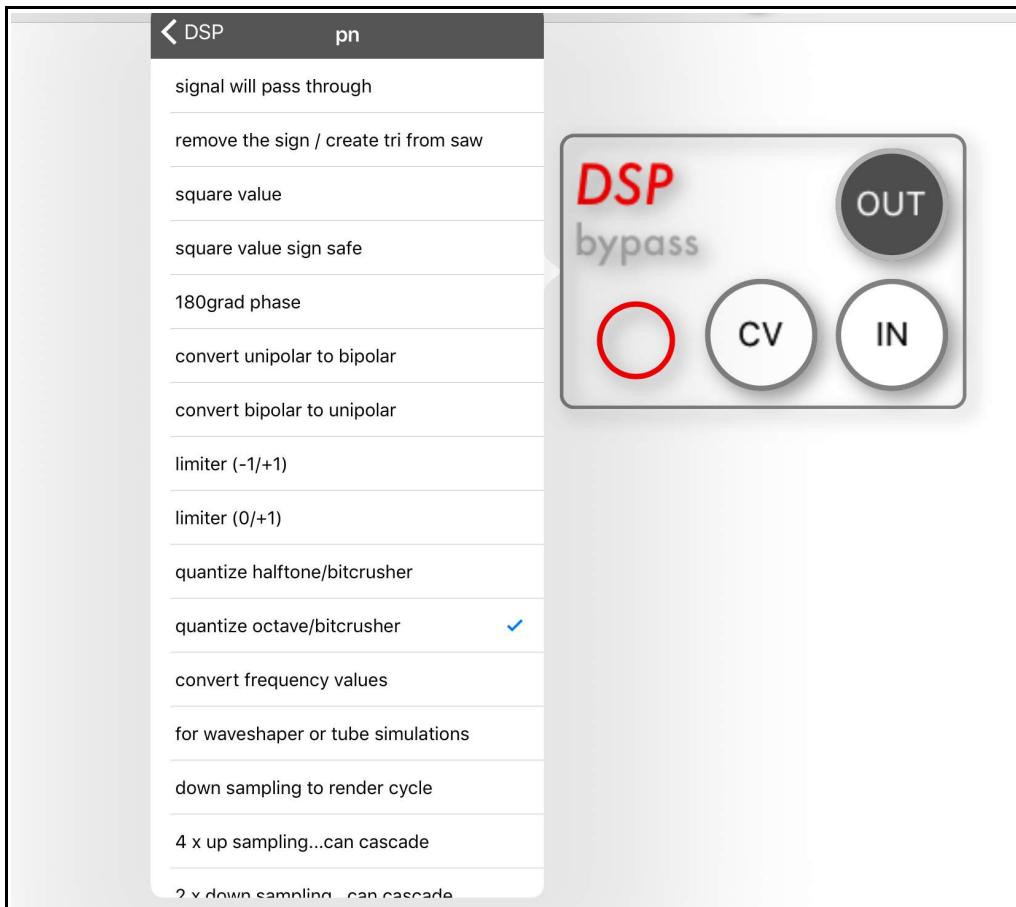
You can select the MIDI channel 1-16 from the bottom section of the Editor.

## Using the MathDSP (DSP) Module

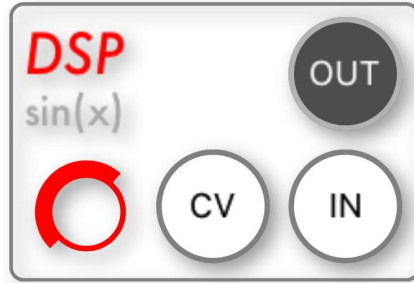
Long tap to access the settings menu, where you can configure the chosen DSP.



Tap on **Pn:** in the Config Settings to access all the available maths and DSP functions (see below).



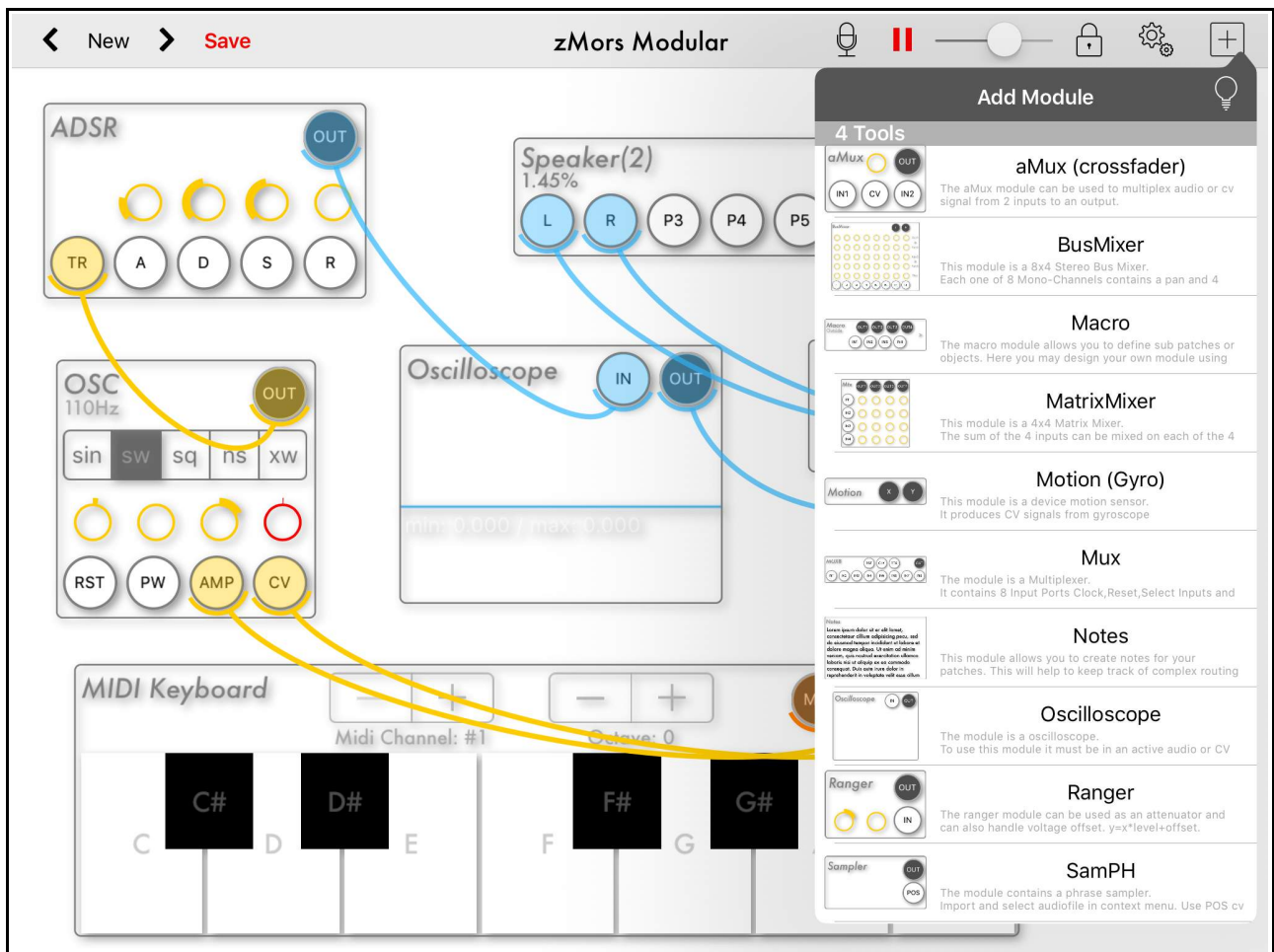
Scroll down through the list to see all the available options.  
You can also just turn the red dial to change the Math/DSP options.



## Creating A Macro Module

It is possible to create Macro Module of the current setup.

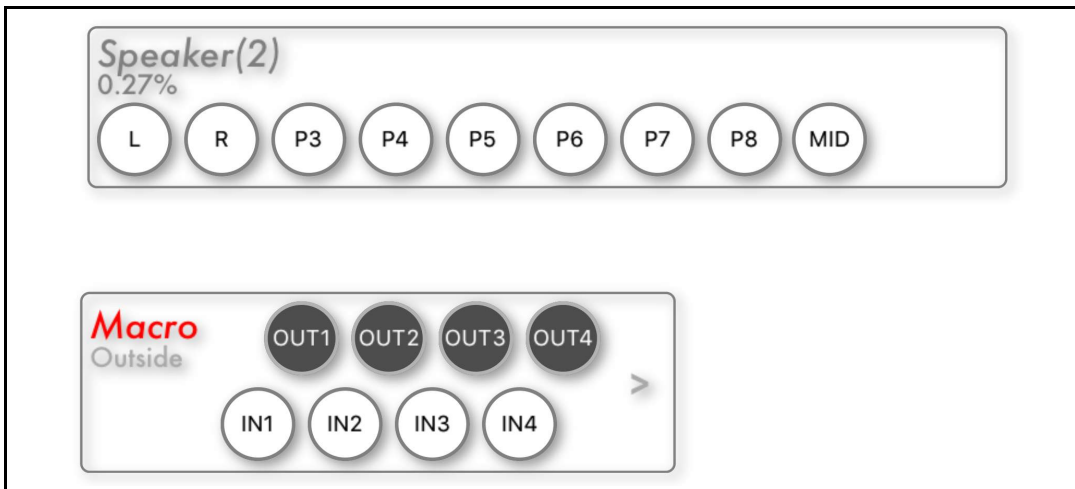
Simply choose the Macro module from the Tools section of the Module List.



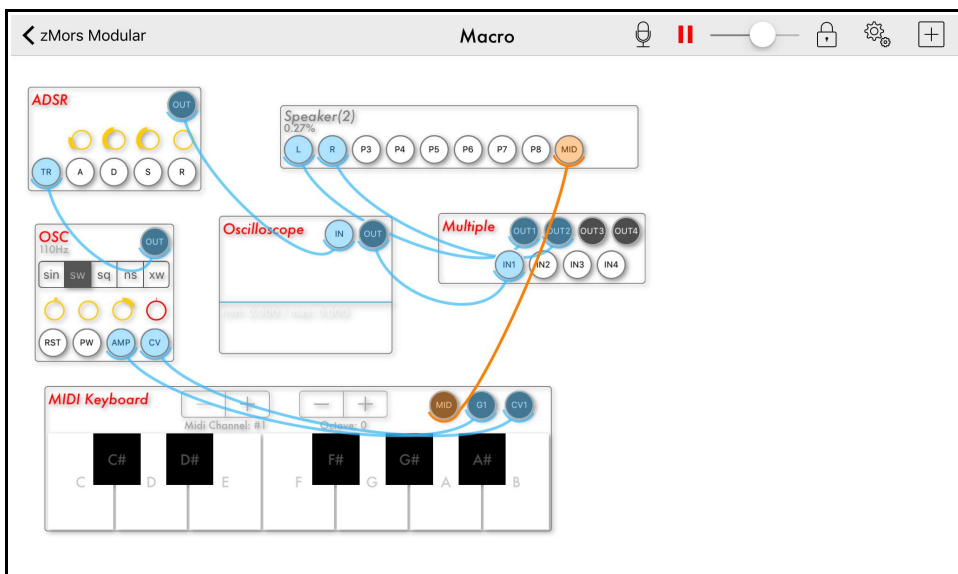
You will next be asked to confirm this action by selecting **Yes** or choose **No** to cancel.



Now you have a new **Macro Module**, which can be used just like any other module.

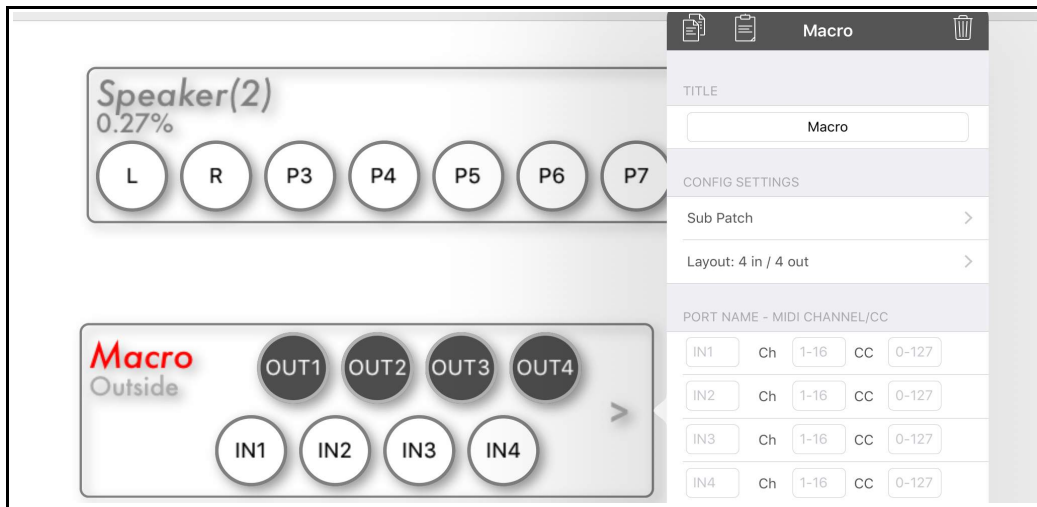


**Tapping** on the arrow icon will give you access to the contents of the **Macro** module again.

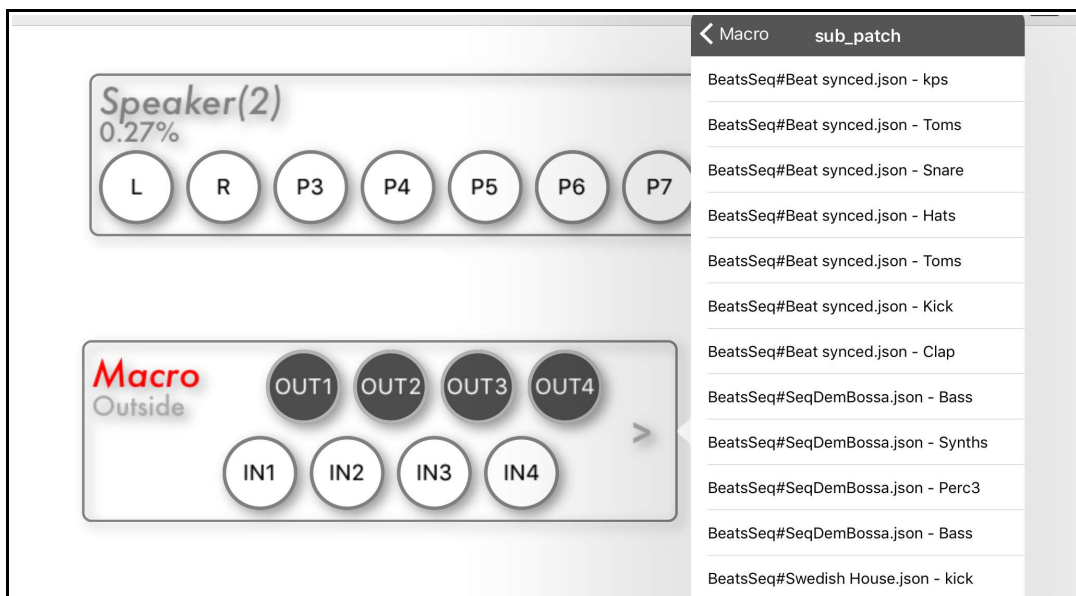




**Tap & Hold** on the new **Macro** Module to access its settings menu.

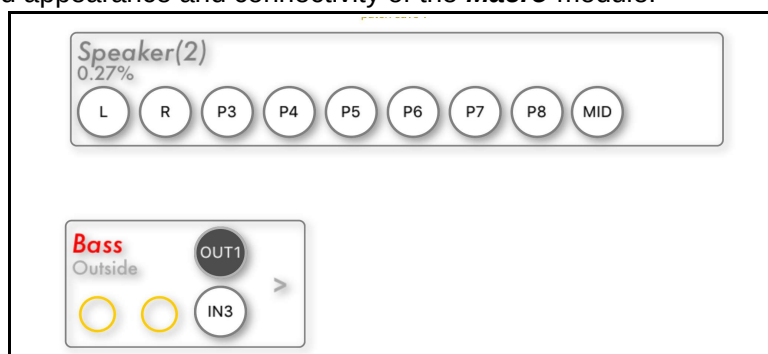


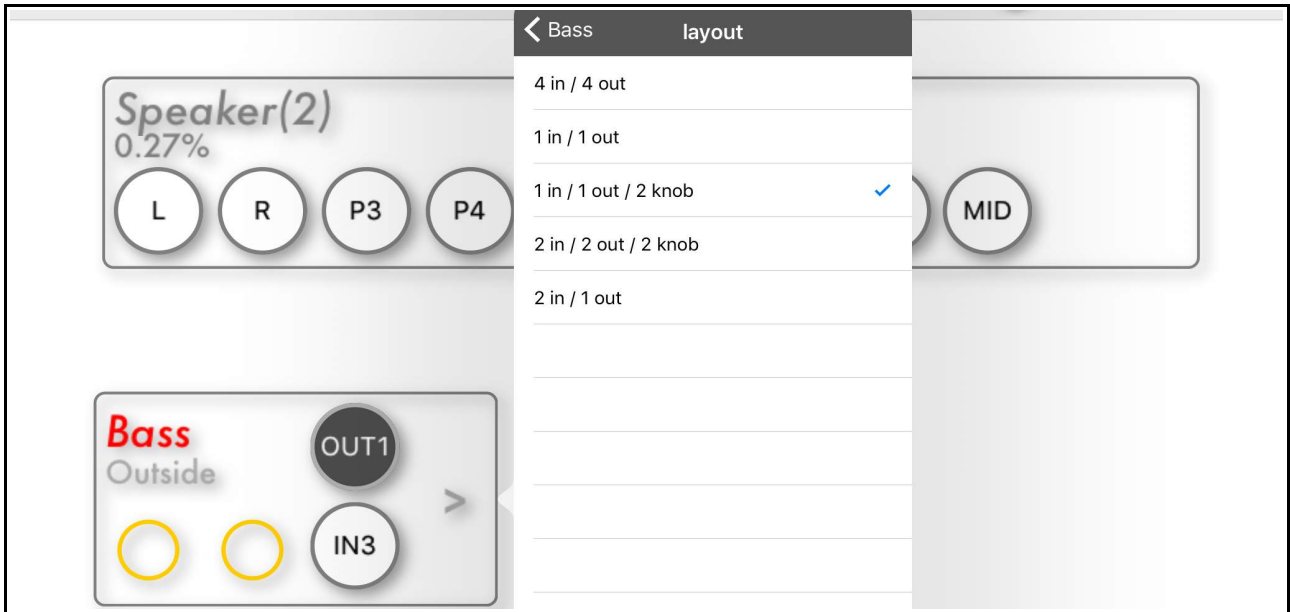
Use the **Sub Patch** option above in the Macro menu to access the **Sub Patch Preset List**. The Sub Patch presets are a list of all the occurring patches within the presets. It is a good idea to create your own because they are easy to re-use in new presets.



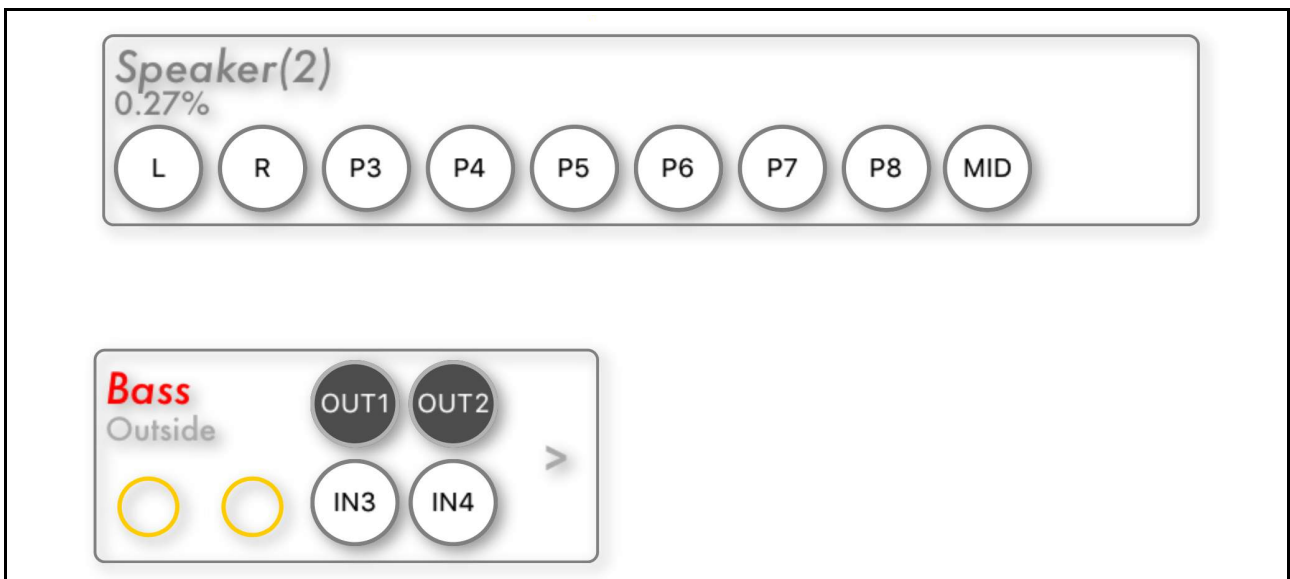
If you select a new preset (for example – Bass), then the Sub Patch will be replaced with your choice.

If you select the **Layout** option from the Settings Menu (shown on the previous page), you are able to change the outward appearance and connectivity of the **Macro** module.





For example – choose **2 in / 2 out / 2 knob** option.



You now have **stereo ins and outs** on your new module.

# THE MODULES

## I - GENERATORS

### OSC (8Bit)

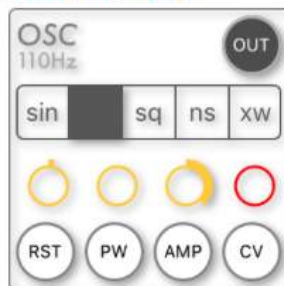


The module is an 8Bit OSC.

You can set a programm, level and downsampling by tap&hold or cv inputs.

Config params: prg, level, downsampling

### OSC (LFO/VCO)



This module is an oscillator (VCO) and can be used as LFO too.

The first CV input will force a Hard Sync by resetting the wave phase, when triggered by the value defined on the first knob. The PW input controls the Puls Wide in square mode, in noise mode a right value produce S/H, in xsaw mode PW will control the supersaw detune.

AMP will control the amplitude of the signal. A left value of the knob will invert phase. The CV knob controls the oscillator frequency. You can use square\_swing when using as LFO trigger for beat module to produce a swing. When the additive sin parameter is set to 1 in sin mode, osc produce 16 harmonic overtones. Spectrum can be controlled by PW input & knob.

Config params: tune, frequency, volume, wave\_mode, triggerlevel, pulsewide, square\_swing, additive\_sin, atn\_am\_mod, atn\_frq\_mod, atn\_pwm\_mod, atn\_rst, init\_phase

### OSC (PhysMod)

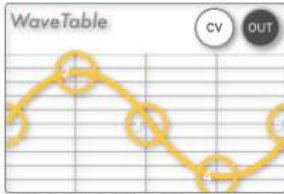


This module is a karplus strong algorithm based oscillator.

CV control the frequency (buffer size). TR will sample the initial wave table. A pre filter and post filter with mod-cv input will shape the wave form in each phase. When WT input is connected to wave table module the initial wave can be additive, otherwise the internal noise generator will be used.

Config params: pre\_filter\_loops, tune, level, filter\_pre, filter\_post, filter\_speed, wavetable\_size, wave\_init\_level

## WaveTable



This is a wavetable oscillator or lookup table.

An input value of range -1/+1 will be translated into a new value defined by the curve. Tap and hold in a head to select preset wave forms. The algorithm is based on gaussian elimination or additive sin gen. Values higher then 1.0 will be clipped.

In oscillator mode (default) you can define tune in context menu and use input as cv pitch select. If input is not connected the output is one phase of buffer size. This is usefull with pm-oscillator module. With XY Pad on input custom waves can be defined. Double tap on drag buttons will change curves.

Config params: preset, additive, unipolar, osc\_mode, tune

## Sampler



The module contains a phrase sampler.

Use AudioPasteBoard to fill it with a short sample or loop or import/select audiofile in context menu. Use TR input to trigger the sample. Use POS input to set start position (unipolar).

Config params: loopMode, fileName, startPosition

## SawPhaseMod



This module is an saw/sin oscillator with phase distortion. CV input controls the frequency. PH controls the phase distortion. You can also set the output to  $\text{dsp\_fastsin}((x*x-1)*(x*x-1)-0.5)*2$ . This module is useful for FM stuff

Config params: tune, frequency, volume, triggerlevel, fastsin, init\_phase

## 2 - ENVELOPES

### ADSR (ENV)

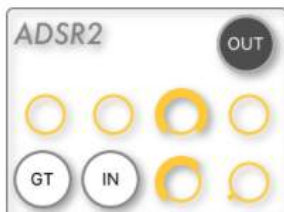


The module is Attack Decay Sustain Release envelope generator.

The parameters can be controlled by knob and/or CV inputs. In the context menu you can define the curve mod (linear=0,exp=1).Set the Output Level to a negative value (e.g. '-1') to invert the envelope.

Config params: attack, decay, sustain, release, trigger\_level, exp\_mode, output\_level

### ADSR2 (ENV)

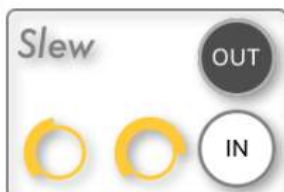


The module is Attack Decay Sustain Release envelope generator.

2 coefficients controls the attack and decay/release curve. It has an input gate and a build in VCA via input port. In the context menu you can set the time base, trigger level and retrigger option.Time Base is a maximum time for ADR parameters in seconds.

Config params: attack, decay, sustain, release, trigger\_level, coef1, coef2, time\_base, retrigger

### Slew



The slew module can be used as a simple filter or envelope generator. With short attack/decay values you get a very simple LowPass filter. Insert this module in CV frequency signal way to add portamento. The algorithm is based on approximation.

Config params: attack, decay

## 3 - PROCESSORS

### Combiner



This module will combine 2 inputs to one output.

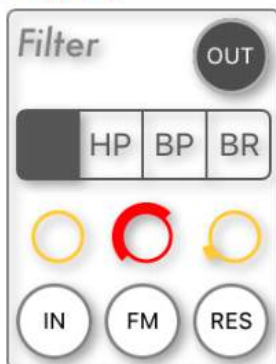
Functions may be selected by using the knob. When an input is not connected you can define a static value in the context menu.

Function list:

- 0:  Sum/Mix  ..... out=in1+in2
- 1:  Mul/Ring  ..... multiply / ringmodulation
- 2:  IN1  ..... solo in1
- 3:  IN2  ..... solo in2
- 4:  Dist  ..... distance
- 5:  Sub  ..... subtract
- 6:  Min  ..... minimum value
- 7:  Max  ..... maximum value
- 8:  ProgMix  ..... out=in1+in2-(in1\*in2)
- 9:  Shaper  ..... out = 1.1f \* in - 0.2f \* in \* in \* in
- 10:  3D  ..... out = dist(in1,in2)
- 11:  lastchange  ..... out = lastchange(in1,in2)
- 12:  threshold  ..... if(in1>in2)out=in1 else in2
- 13:  gate  ..... if(in1>in2)out=in1 else 0
- 14:  iclip  ..... inverse clipping
- 15:  pmod  ..... phase modulation
- 16:  simson  ..... Simpson integration
- 17:  vrsun  ..... sum(in1)\*in2
- 18:  vindex  ..... idx(in1,table(in2))
- 19:  vtmerg  ..... vtmerg
- 20:  midimerge  ..... merge 2 midi signals
- 21:  Off  ..... no rendering

Config params: in1, in2, prg, alpha

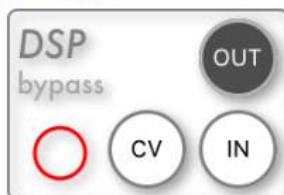
### Filter(VCF)



This module is a multimode filter with LowPass, HighPass, BandPass and BandReject. You can define frequency and resonance via knob and CV inputs. The LowPass is a Moog-style filter and will render sample accurate. So very fast modulations are possible but this uses more CPU. The HP,BP,BR mode are DSP based. That means a change of CV input is Sample/Hold based and depends on current buffer size and scale in global settings menu. Current buffer size and scale can be found in global settings menu.

Config params: drive, frq, res, atn\_frq\_mod, lp\_limit, lp\_r1, lp\_r2, lp\_r3, lp\_r4

## MathDSP



The module contains DSP Math functions.

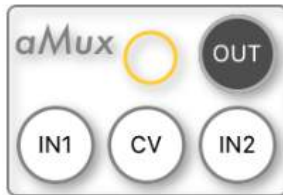
Functions may be selected by using the knob. This module is optimized for very low cpu usage. Use it a lot ;- ) but avoid feedback loops when using up/down sampling!

Function list:

- 0:  bypass  ..... signal will pass through
  - 1:  abs(x)  ..... remove the sign / create tri from saw
  - 2:  x\*x  ..... square value
  - 3:  [x\*x]  ..... square value sign safe
  - 4:  invert  ..... 180grad phase
  - 5:  Uni2Bi  ..... convert unipolar to bipolar
  - 6:  Bi2Uni  ..... convert bipolar to unipolar
  - 7:  ClipBi  ..... limiter (-1/+1)
  - 8:  ClipUni  ..... limiter (0/+1)
  - 9:  Quant 12  ..... quantize halftone/bitcrusher
  - 10:  Quant Oct  ..... quantize octave/bitcrusher
  - 11:  pow2(x)  ..... convert frequency values
  - 12:  tanh  ..... for waveshaper or tube simulations
  - 13:  DownSampl  ..... down sampling to render cycle
  - 14:  4xUpSampl  ..... 4 x up sampling...can cascade
  - 15:  LoFi  ..... 2 x down sampling...can cascade
  - 16:  conv:55Hz/V  ..... convert CV voltage to korg standard
  - 17:  DCfix  ..... fix DC level on complex signals
  - 18:  fract(x)  ..... truncate to fraction
  - 19:  1-x  ..... invert unipolar signal
  - 20:  eShp  ..... shaper, use 'eShp drive' 1-64
  - 21:  meamgv  ..... mean magnitude from audio 2 cv
  - 22:  slideLP  ..... very fast low pass 'slide lp' 0.001-0.999
  - 23:  sin(x)  ..... compute the sin(x\*2\*pi)
  - 24:  log2(x)  ..... compute the log2(x)
  - 25:  log10(x)  ..... compute the log10(x)
  - 26:  DsRamp  ..... downsampling with ramp
  - 27:  pitch2cv  ..... compute pitch cv by zerocrossings
  - 28:  maxmgv  ..... compute max peak per cycle
  - 29:  cyclePeak  ..... compute single sample peak per cycle
  - 30:  mono2poly  ..... use to inject into polypatch
  - 31:  mute  ..... no output / keep input in sync (AB compare)
  - 32:  rmsqv  ..... root mean square
  - 33:  saw  ..... alias free saw
  - 34:  off  ..... no output / no input compute (AB compare)
- Config params: pn, slide\_lp, eShp\_drive

## 4 - TOOLS

### aMux (crossfader)



The aMux module can be used to multiplex audio or cv signal from 2 inputs to an output.

Config params: pan

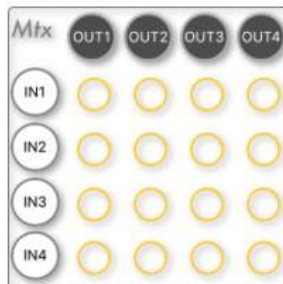
### Macro



The macro module allows you to define sub patches or objects. Here you may design your own module using nested modules. You have 4 inputs and 4 outputs to route through the Macro module. You can select recent patches and the layout by context menu. To edit a sub patch, press the right '>' button.

Config params: sub\_patch, layout

### MatrixMixer



This module is a 4x4 Matrix Mixer.

The sum of the 4 inputs can be mixed on each of the 4 outputs buses.

Config params: bipolar, I00, I01, I02, I03, I10, I11, I12, I13, I20, I21, I22, I23, I30, I31, I32, I33

### Motion (Gyro)



This module is a device motion sensor.  
It produces CV signals from gyroscope

### Mux



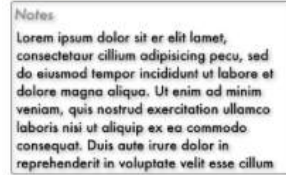
The module is a Multiplexer.

It contains 8 Input Ports Clock, Reset, Select Inputs and a output. When step select is connected, Clock & Reset input are disabled. Render all will keep all inputs in sync but needs a lot of cpu. Step Select can be configured in binary mode then output is summing one or more inputs.

Config params: render\_all, sts\_binary, current\_step



## Notes



This module allows you to create notes for your patches. This will help to keep track of complex routing concepts as you build them.

## Oscilloscope



The module is an oscilloscope.

To use this module it must be in an active audio or CV chain, with connections routed through the In and the Out jacks. You can select mode by context menu.

Config params: scale\_y, scale\_x, cycle\_sync, audiogramm

## Ranger



The ranger module can be used as an attenuator and can also handle voltage offset.  $y=x*\text{level}+\text{offset}$ .

Config params: level, offset

## Split (Multiple)



This module is a multiple.

The sum of the 4 inputs will be output on each of the 4 outputs. Use this module to split incoming signals across multiple destinations. The render cycle is loop safe. When you create feedback loops you can get signal delays, depending on the buffersize/scale selected in the global settings.

Config params: layout

## VCA

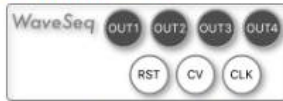


This module is a voltage controlled amplifier.

Exp/Linear mode can be switched in the context menu. With cpu saver the render input stops while cv input = 0.

Config params: in, cv, exp\_mode, attenuation, cpu\_save

## WaveTable Seq



This module allows you to create granular synths. The generator produces a saw signal on 1 output. When phase is done or clk input is active the output will change to next port. Try the Granular and Wave Jumper preset.

Config params: tune, frq\_mod, volume, binary\_mode, phase\_count

# 5 - SEQ & CV

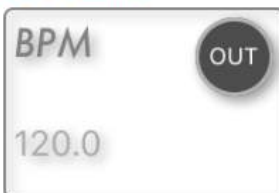
## Ableton Link Clock



This module is an clock for Ableton Link. RST will trigger based on quantum value. CLK will trigger on each 1/16 note. The pulsewidth parameter is used for CLK and RST

Config params: quantum, beatsPerClick, pulsewidth

## BPM / Clock

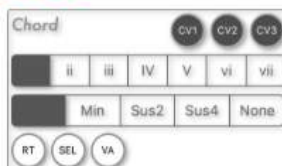


The module contains BPM clock gen functions.

The output signal is a unipolar saw or MidiClock. Use Combiner in gate mode to create delayed triggers.

Config params: bpm, midiClockMode

## Chords

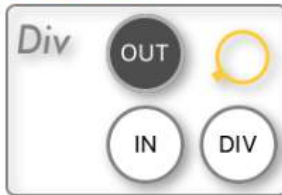


The module is a chord generator.

This module will generate chords, via the three CV Outputs. Each of these will output a note in the selected chord. You can select a chord (I, ii, iii, IV, V, vi, vii) using the module buttons or via CV. Each can be modified to a Maj, Min, Sus2, Sus4. When 'none' is selected the root CV will be applied on all 3 Outputs. Chord selection can be controlled by input SEL, and modify modes are selected with VA. The RT Input will change the root key.

Config params: chord\_number, chord\_var

### Clock Divider



This module is an Clock Divider. Divide by (1-16) and trigger level can be selected by knob and cv.  
Config params: div, trigger\_level

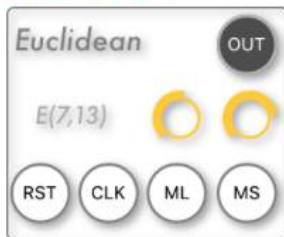
### CV Sequencer



The module is a analog style 8 Step CV Sequencer.  
You have a reset, clock and analog step select input. The output values depend on the knob values.

Config params: cv1, cv2, cv3, cv4, cv5, cv6, cv7, cv8, triggerlevel, add\_to\_input

### Euclidean Seq



This module is an Euclidean sequencer. Pattern length and step count can be selected by knobs and mod inputs ML,MS. CLK and RST inputs can be used for syncing.  
Config params: length, steps, trigger\_level

### Quantizer



This module is a Quantizer. The first 12 bits of bitmask represent halftones. When all bits unset then quantization is disabled

Config params: bitmask

### Sample&Hold



This module is a Sample&Hold or Tap&Hold module

Config params: tap\_and\_hold, trigger\_level

## Slider



The slider module is a simple slider.  
The default range is -1/+1. The range can be changed in the context menu.

Config params: min, max, cv

## Step Sequencer



The module is a 5 track gate/trigger sequencer.  
When you activate 2 sequential steps you will get a single long impulse on the output, depending on the pulse sample parameter.  
The SAW output produces a stair-step waveform with a frequency of one bar. You can use it with slew+wavetable to create automations. The STM CV Input will modulate the current step position. The GT input will control the gate time. When the context menu parameter `iaa_sync` is 1 and an IAA connection is active, the Inter-App Audio Host will override the beat clock. You can override the sample offset too.

Config params: triggerlevel, triggerlevel1, triggerlevel2, triggerlevel3, triggerlevel4, pulse\_sample\_count, sample\_offset\_iaa, iaa\_sync, step\_count

## XYPad

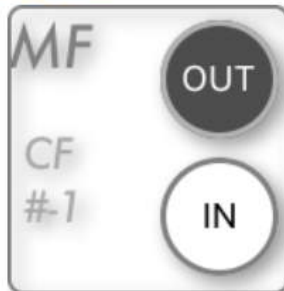


The module is a simple XY Pad.  
Each axis has two outputs. Ranges can be configured in the context menu. Values are limited by a range of -4/+4. The quad value controls the curve 0=linear/1=exponential. When the XYPad is touched a trigger is sent on the TR output.

Config params: x1\_min, x1\_max, x1\_quad, x2\_min, x2\_max, x2\_quad, y1\_min, y1\_max, y1\_quad, y2\_min, y2\_max, y2\_quad

## 6 - MIDI

### Midi Filter

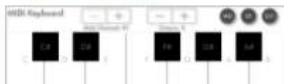


This module is a Midi Tool.

You can filter only a midi channel and/or extract midi cc as cv and notes as trigger. When channel > 0 and cc2cv=0 output is midi filtered by channel number. When cc2cv>0 output is unipolar cc message or velocity when cc2cv is a note on/off message.

Config params: operation, channel, show\_midi\_messages

### Midi Keyboard

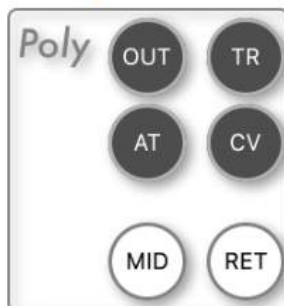


This module is a one octave MIDI keyboard.

1 Gate and CV Output allow the module to play. Midi input messages are mapped to midi output also. Midi velocity will be mapped to the gate values. Midi Output produces a digital signal that can be decoded with PolyLoop and midi filter module. The octave stepper changes the octave range. The current midi channel can be defined by context menu. CV outputs operate in volt per octave. The range is -10volt/+10volt. 1 volt is 0.1 in float, 2 volt is 0.2 in float and so on.

Config params: cv1, g1, octave, midi\_channel, midi\_sources

### Midi Poly

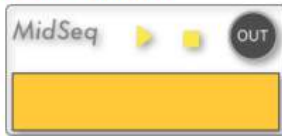


This module is a Polyphonic Loop.

All modules in the loop (tr,cv,at -> ret) will be cloned up to 16 times. When using modulations from outside the loop use dsp in mono2poly mode. You can use all modules excluding: beat,bpm,verb,delay in the subchain. The Mid input port will receive digital midi data from e.g. keyboard and compute midi note on/off, velocity, cv and after touch. You can use midi filter module to filter midi channel or extract any cc messages. Red cables in the subpatch signaling polyphonic compatible modules.

Config params: max\_poly, level, transposition

## Midi Sequencer



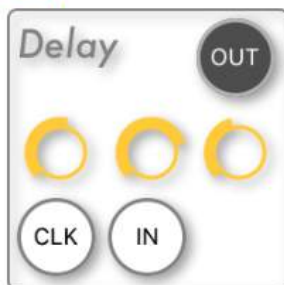
The module is a midi file Player.

You can import a midi file from app documents directory or icloud files. Use midi filter module to extract midi channels and connect to polyphonic module to create a complex sequence.

Config params: Import Midi File, bpm, track\_length, note\_length, cycle\_start, cycle\_length, cycle\_active, midiEvents, swing

## 7 - FX

### Delay

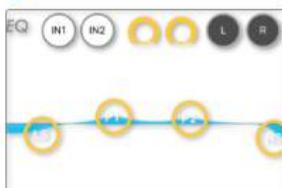


The module is a sample delay.

The left knob controls the delay time from 0 to 4 seconds and the middle knob controls the dry/wet ratio. The feedback value can be controlled by right knob.

Config params: dly\_time\_s, drywet, feedback, bpm\_sync

### EQ



This module is a 4 Band Stereo EQ.

Filters are: low shelf,peak,peak,high shelf

Frequency and gain can be controlled by touch buttons and the q- value for the 2 peaks by knobs.

Double touch on a touchbutton to reset frequency & gain.

Config params: f0\_f, f0\_q, f0\_g, f0\_t, f1\_f, f1\_q, f1\_g, f1\_t, f2\_f, f2\_q, f2\_g, f2\_t, f3\_f, f3\_q, f3\_g, f3\_t, f4\_f, f4\_q, f4\_g, f4\_t, f5\_f, f5\_q, f5\_g, f5\_t, f6\_f, f6\_q, f6\_g, f6\_t, f7\_f, f7\_q, f7\_g, f7\_t

## InterAppAudio



This module is a inter-app-audio host.

When using a instrument node you can use MID as midi input. When using a effect node you can use IN1+IN2 as stereo audio input. If hanging on start, try to start the other app in the home screen and quit.

Config params: IAA\_FX

## PureData



The module contains a pure data wrapper (vanilla).

You can select any puredata patch files. The module support 2 audio inputs [adc~], 2 audio outputs [dac~] and midi [notein|ctlin] note and midi cc messages. Remember you can only use a single instance in a patch

Config params: fileName

## Reverb



The module is a reverb based on GVerb library. Thanks to Juhana Sadeharju!

The left knob controls dry/wet ratio. The DW port can modulate the dry/wet ratio too. The right knob selects a reverb algorithm preset. Use this effect as an insert and avoid loopbacks.

Config params: roomsize, revtime, damping, spread, inputbandwidth, earlylevel, taillevel, drywet

## Tape/LoFi



The module contains sampling rate transform.

Use SR knob or cv input to modify the input sample rate for tape stop effects. Use TI knob or cv input to get a LoFi effect

Config params: lofi, srf

## 8 - GUI

### Console



This module is a mixing console with 8 channels.

Each channel contains knobs for level, pan, aux1 & aux2 sends and buttons for a generator plugin and effect macro, solo/mute, midi channel select and mono/stereo. This module based on 8 instruments macros, 8 fx macros, 1 aux macro with 2 stereo busses and 8 midi channel filter modules. Channel Metering indicate current Level or a red dot when sub-renderer is going off by no midi activity.

Config params: auto\_gate

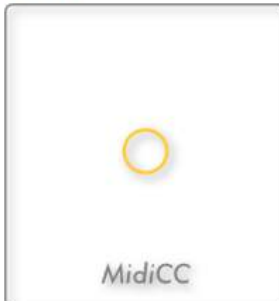
### JSCtrl



The module is a controller.

It contains a keyboard, volume fader, button lines and XY pads. It is full customizable with by javascript. A JavaScript source code editor with api reference is included.

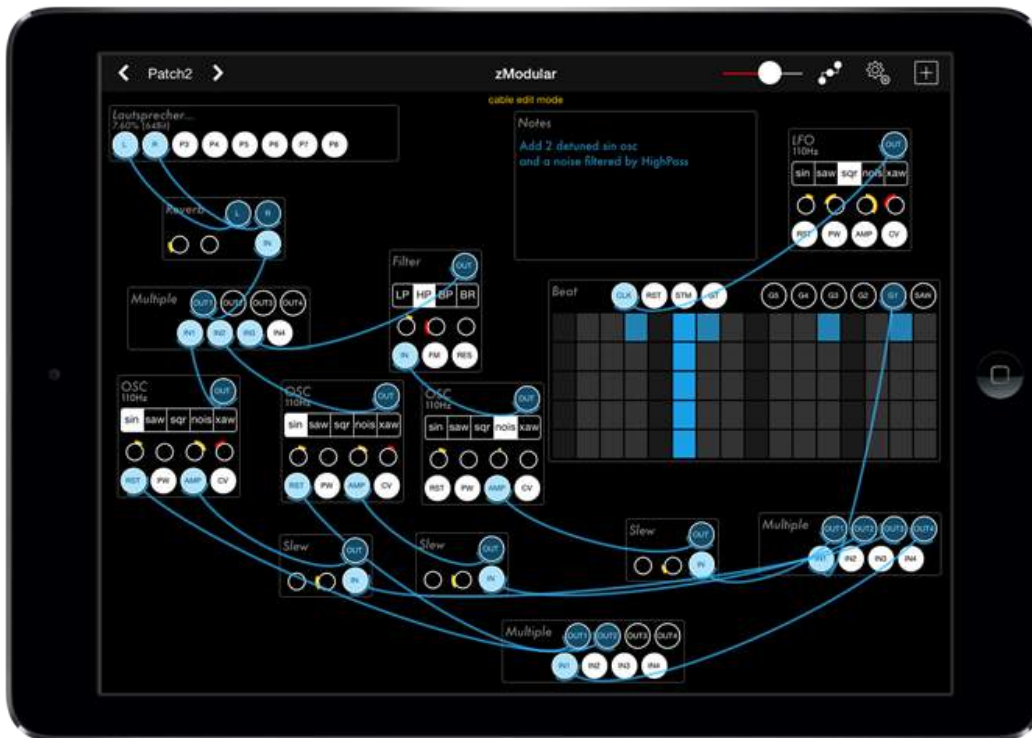
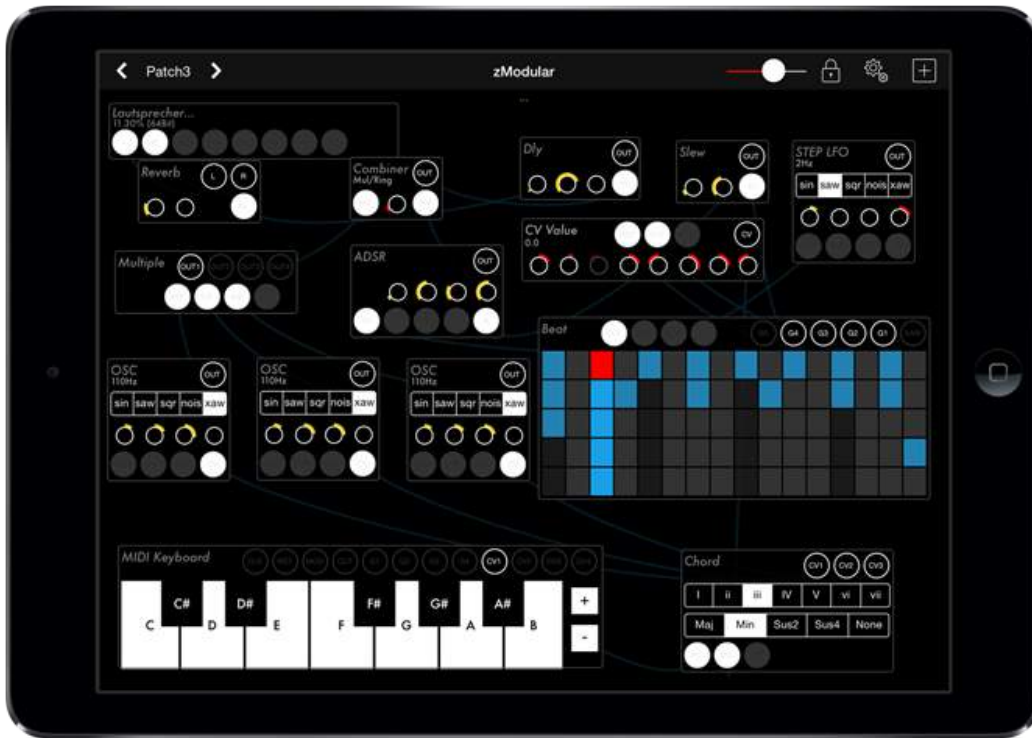
### MidiCC



This is a single knob with midi cc send. Midi channel can be defined by ch (0=off) or 1-16.

Config params: cc, ch, val





Manual by Nick Trass