Important Changes (Version 2.0)

This major release of **Stria** is completely compatible with Stria 1, you can play old presets.
You can enable “Include Control Manager in Presets” -in the Settings- to save the current assignations (MIDI, Scrub Y, Accelerometer and LFO). This is disabled by default to keep compatibility with version one.

WARNING!
If you enable the “Include Control Manager in Presets”, the old presets will load without any assignments. This will cause the cancellation of all global “Control Manager” assignments.
You should save a new preset (tmp) with current global assignments, thus as overwrite some Presets to include the “Control Manager Assignments”.

The new Pad (i.e. Pad 2) of this version of Stria, is enabled to receive MIDI ‘note On/Off’ messages. Now you can play the second Pad through MIDI Keyboard remotely. See ‘Midi Manager’ for more details.

Introduction and Acknowledgments

“Stria” is the title of a very important composition by [1]**John Chowning** and I would like to say that this computer Music piece is the manifest of Frequency Modulation technique applied to audio synthesis, invented by Mr. Chowning himself. *Stria* app is a humble tribute to that great person and also to [2]**Max Mathews** who was the inventor and (grand) father of the well known Music-n sound synthesis language generation (*Stria* engine is **Csound** for iOS based). Among the different implementations of this type of language, Csound (developed for the first time by **Barry Vercoe** at MIT Media Lab), is certainly one that has grown enormously in recent years. Thanks to **Victor Lazzarini** and **Steven Yi** for Csound API for iOS. At the same time, *Stria* was musically inspired by some interesting ideas about the use of tuning systems. I am deeply grateful to [3]**Walter Branchi**, whose great knowledge of the tuning

[1] **John M. Chowning** (/ˈtʃaʊnɪŋ/; born August 22, 1934 in Salem, New Jersey) is an American composer, musician, inventor, and professor best known for his work at Stanford University and his invention of FM synthesis while there. (Wikipedia)

[2] **Max Vernon Mathews** (born November 13, 1926 in Columbus, Nebraska, USA – April 21, 2011 in San Francisco, CA, USA) was a pioneer of computer music. (Wikipedia)

[3] **Walter Branchi** (Roma, ...) è un compositore, musicista, scrittore e giardiniere italiano. È noto per aver preso parte al Gruppo di Improvvisazione Nuova Consonanza tra il 1966 e il 1975 e per la coltivazione di rose antiche francesi dell'Ottocento[1][2] in un giardino-vivaio di particolare valore[3] in Provincia di Terni. (Wikipedia)

systems has prompted to me the main guidelines of this project. This work is also dedicated to him.
Eugenio Giordani

Overview

Stria is essentially a sound synthesizer working in real time, the generated sound is mainly composed as a time continuous bands of frequencies using concurrently types of sound synthesis and manipulation:

a) Simple FM synthesis
b) Additive synthesis
c) Granular-like synthesis concepts
d) Wavetable synthesis basis
e) Subtractive-like synthesis

We can think *Stria* as a virtual bank of oscillators and relative amplitude envelopes, generated continuously and overlapping in time (*note blast*). A controlled timer triggers “notes” (in a granular fashion) using two main parameters (for both Pads):

Event Rate (rate at witch notes or grains are generated)
Event Duration (time lasting a single note or grain object)

1) Volume
2) Rate
3) Duration
4) Spread
5) FM Index
6) Wave Source Morph

The main Pad (i.e. Pad 1) performs advanced Mixture:

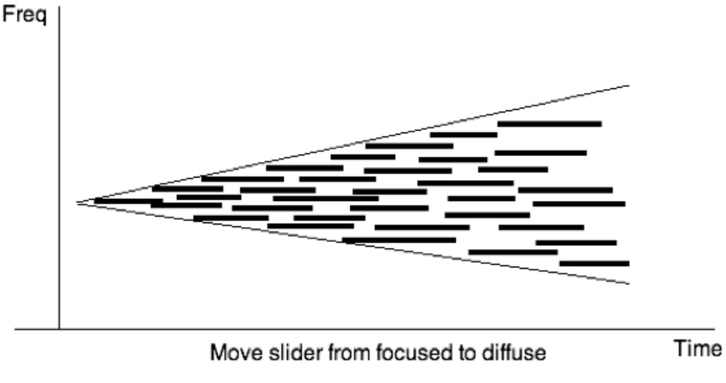
7) Mixture Components
8) Mixture Grid Step

Volume, rescales the amplitude of Events, this does’t affect current Note Event, but the next events;

The faster the Rate parameter, the greater the number of overlapping notes is generated. Similarly, the amount of overlaps also depends on the note duration. The maximum level of overlap is reached when the **Event Rate** slider is on the Fast position (turned completely right) and the **Event Duration** is on the Long position (turned completely right). You can move both the sliders during the performance in order to obtain the desiderated degree of complexity. The ranges of those two parameters are as follows:

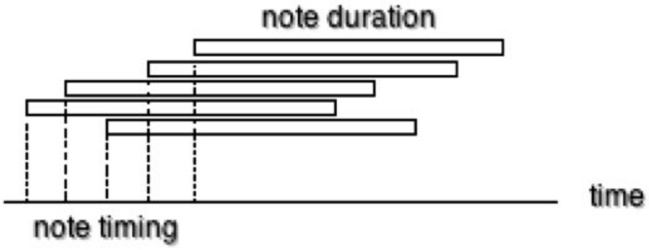
a) **Event Rate**: [0.25 ÷ 10] gps i.e. grains for second
b) **Event Duration**: [0.1 ÷ 4.0] seconds

At the minimum rate of 0.25 sec the engine generates 1 note every four seconds (e.g. 1/0.25 = 4) and at the opposite value ten notes per second are produced.
The basic frequency of the note blast is controlled mainly by the position of the horizontal component (X axis) of the own **Pad** on the virtual control surface. The position of the X in the **Pad** controls the deterministic component of the base frequency of every generated note in the blast. All the notes generated keep the base frequency value throughout their duration, the actual frequency of each note.
The actual frequency of each note is formed by two pitch components: a *deterministic* part plus a *random* part, the first of which, as stated above, is controlled by the X axis of pad. The *random* component of the base frequency is then controlled by the **Frequency Spread** (slider control in the *Stria* Parameters). When the **Frequency Spread** is at minimum value (Focused), all the notes generated have almost the same frequency (not exactly the same). As you move the slider to the right position (Diffuse), each new generated note will have larger frequency displacement in respect to the base deterministic value set by the X control on the **Pad**. For example, if you start with a simple sine wave at frequency f_0 and gradually move the slider control from the focused to diffuse position, you will obtain a gradual increase of a frequency band centered on f_0 as sketched in figure:



This is the basic concept on how *Stria* works. The multiple generation of overlapping notes and its frequency dispersion produces narrow bands of frequencies that are time uncorrelated so the resulting sound is very rich internally and almost never boring. One can move through different degrees of band thickness, from a steady pitched sound to a fat chorus-like sound and vice-versa.

Every single note is amplitude controlled by a smooth symmetrical **Event Envelope** (Pad 1 and Pad 2) of which you can control the *attack* and *release* time. In order to obtain a smooth sound you have to set this parameters at the maximum level.

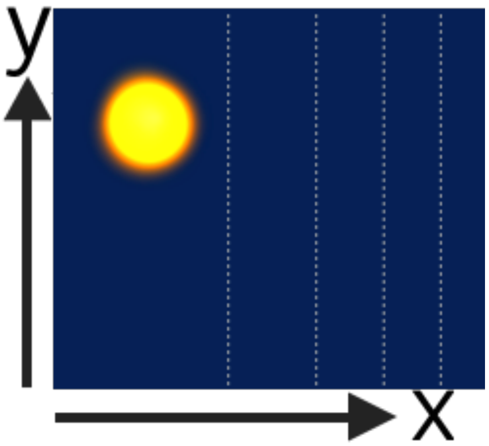


In figure is shown the disposition in time of a sequence of notes of a certain duration triggered in time and overlapping one to each other. The level of overlap depends on the ratio of note duration and the rate of note generation.
In the next pages we will discover many other interesting aspects and possibilities of *Stria*, starting from the **Spectral Grid** pitch system management to the timbre transformation.

Event Pitch Jitter (Pad 1 and Pad 2) superimpose at current note pitch a random offset (+/- bipolar), you can manage the range of random generator through **Delta** parameter expressed in semitones and the frequency of random generator by **Random** (Hz). Unlike **Frequency Spread** this transposes up/down the entire mixture (depending from **Mixture Components** in *Stria* parameters).

Pads Features

This new version (2.0) of Stria, introduces a second Pad (i.e. Pad 2).
As you know, Pad 1 is capable of performing ‘**Complex Mixtures**’, consisting of up to eight FM modules tuned on the ‘**Spectral Pitch Grid**’ for every Event generated by Granular-like synthesis concepts for the parameters ‘Rate’ (i.e. density of events) and ‘Length’ (i.e. duration of events).
Pad 2 instead performs a simple (one module) FM, tuned in the free range 20 Hz ÷ 2.5 kHz. Like the Pad 1 you can manage almost all parameters apart the Event Mixture (Components and Grid Step). Unlike Pad 1, which you can assign the Scrub Y axis, Pad 2 has a pre-assigned Scrub Y. (See Control Manager for more details.)



The scrub Y component of Pad 2's joystick can be considered as ‘Amount’ for these three main Random Parameters below:

- FM Setup Carry Min/max (when !=)
- FM Setup Mod Min/max (when != and Event FM Index > 0)
- Event Parameters Pad 2: Spread (when > 0)

The range is from zero (Y = 0 at the bottom) to one (Y = 1 at the top).
For example, with the ‘Event Parameters Pad 2: Spread’ > 0, the Scrub Y will scale from 0 to current Spread value.

AutoGen Pad 2 (only Pad 2)

You can enable the Auto Events Generation for the Pad 2. The rate generation is depending on ‘Event Parameters Pad 2: Rate’ value and ‘Event Parameters Pad 2: Duration’ for the duration.

The ‘Scale Chooser’ button opens a menu where you can choose among several scales used as ‘frequency grid’ for Pad 2.

∧ ∨ The two arrowheads are used to change the pitch height (octaves).

⊖⊕ Tapping on these icons you can change the pitch range of the Pad: from 1 to 6 octaves.

Spectral Pitch Grid (only Pad 1)

The sound generator has two other important features that make it very interesting and powerful.
The first one is the possibility to generate the sound using two different techniques:

FM Frequency Modulation (FM) Synthesis (both Pads)

WS Additive Synthesis (WS) Wavetable (only Pad 1)

FM synthesis consist of a simple patch of modulating/carrier oscillator set with a modulation index controlled by a prefixed envelope while in the Additive Synthesis we have the possibility to add up to eight components for each generated note. At the same time, you can choose continuously from a set of waveforms obtained with a Fourier summation. In other words, you can make an additive sound using sinusoidal/non- sinusoidal waveform and during the performance you can smoothly move among them.

Read **FM Setup** for more information.

The second feature increases complexity and coherence to the sound generation. You can create your pitch space (**Spectral Grid**) using five different criteria of construction. This determines a pitches grid and all the frequency movement comes from that basic selection.

The control surface contains other four types of pitch grid compiler:

Geometric Pitch Grid Compiler

Harmonic Pitch Grid Compiler

Fibonacci Pitch Grid Compiler

Scalar Pitch Grid Compiler

Tempered Pitch Grid Compiler

N.B. Spectral Grid Pitch will affect only the complex ‘Event Pad 1’

You can create a **Geometric** pitch space based on the ratio of specific pitch interval. For example if you specify **Interval Ratio** of 1.5, you will create a pitch space grid based on the natural interval of 5th. Because you can choose any relationship, you can virtually have infinite pitch spaces. The interval ratio defines a coefficient of multiplication than serves to build up the whole pitch grid while the **Base Frequency** is the starting point from which the grid take its origin. The base frequency for the grid in Hz, is the most left frequency for the X pad index.

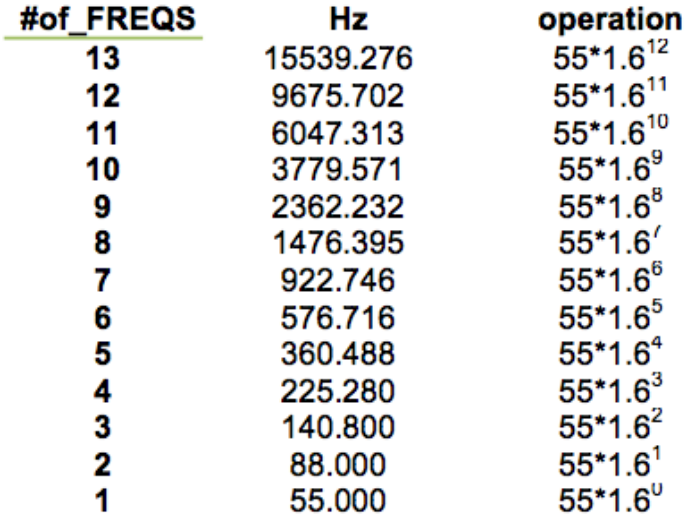
Spectral geometric recursive formula:

$$\boldsymbol{f_i = f_{(i-1)} * iterval\_ratio}$$

where f_i is the i-th frequency
(i.e. f1 = 30 (base) $f_2 = f_1$ * 1.5 = 45.00 * 1.5 = 67.50).

The pitch grid is stored into an internal table up to 2048 location. For smaller values of the interval ratio, the grid will contain more and more frequency (i.e. for interval ratio = 1.038 will be generated 152 frequencies). The program always will stop the grid generation before the actual Nyquist limit is reached (i.e. Sampling Rate / 2).

For example, you will create a geometric pitch grid based on the interval 1.6 that start at frequency A1 (55 Hz). The grid will consists of a total of 13 frequencies starting from 55 Hz up to 15481.1279 Hz as sketched in the next table:



#of_FREQS	Hz	operation
13	15539.276	$55*1.6^{12}$
12	9675.702	$55*1.6^{11}$
11	6047.313	$55*1.6^{10}$
10	3779.571	$55*1.6^{9}$
9	2362.232	$55*1.6^{8}$
8	1476.395	$55*1.6^{7}$
7	922.746	$55*1.6^{6}$
6	576.716	$55*1.6^{5}$
5	360.488	$55*1.6^{4}$
4	225.280	$55*1.6^{3}$
3	140.800	$55*1.6^{2}$
2	88.000	$55*1.6^{1}$
1	55.000	$55*1.6^{0}$

The **Harmonic** pitch compiler, as for Geometric, is located on the tools bar, in this panel are shown all the controls you can use for the purpose to create a certain harmonic pitch grid. In a similar way as the Geometric Grid, you have two main parameter to mange plus a third:

a) **Base Frequency** in Hz
b) **Inharmonic** factor
c) **Harmonics** (number frequencies)

The Harmonic Grid Compiler creates a grid of frequencies according to the natural series of integer numbers (**Harmonics** 1,2,3...etc) starting from a fundamental frequency **(Base Frequency**). In addition you can create *stretched* or *compressed* harmonic spectra putting the **Inharmonic** parameter at a non zero value. Stretched spectrum is obtained for values greater than zero while for compressed spectrum you have to select values less than zero. The stretched or compressed spectra are created using the following formula:

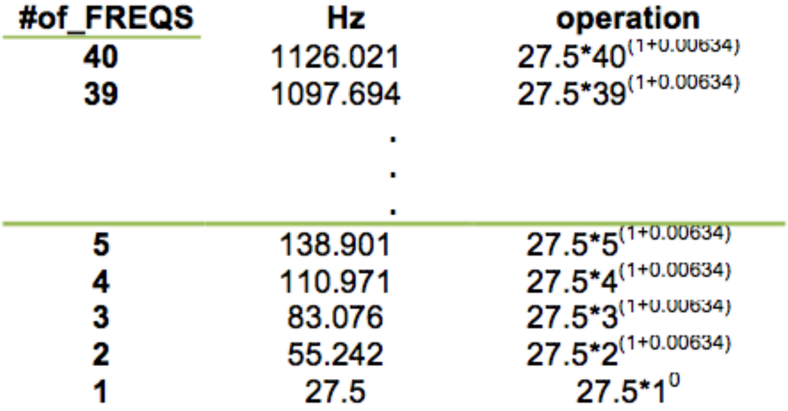
$$\mathbf{F_n = F_{0 * n} (1+exp)}$$

where:
Fn = n-th frequency
F0 = fundamental frequency (**Base Frequency**)
n = index of Fn frequency
exp = factor of expansion/compression (**Inharmonic**)

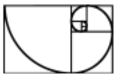
For example you will create an harmonic pitch grid of 40 frequencies based on the series of natural numbers start at frequency A0 (27.5 Hz).

#of_FREQS	Hz	operation
40	1100.0	27.5*40
39	1072.5	27.5*39
	.	
	.	
	.	
5	137.5	27.5*5
4	110.0	27.5*4
3	82.5	27.5*3
2	55.0	27.5*2
1	27.5	27.5*1

The Inharmonic factor can vary from -0.01 to 0.01. When Inharmonic = 0 the generated frequency grid is pure harmonic. Using the same parameter for the previous table but with INHARM = 0.00634 the grid will result as follows:



#of_FREQS	Hz	operation
40	1126.021	$27.5*40^{(1+0.00634)}$
39	1097.694	$27.5*39^{(1+0.00634)}$
	.	
	.	
	.	
5	138.901	$27.5*5^{(1+0.00634)}$
4	110.971	$27.5*4^{(1+0.00634)}$
3	83.076	$27.5*3^{(1+0.00634)}$
2	55.242	$27.5*2^{(1+0.00634)}$
1	27.5	$27.5*1^{0}$

 The third mode for generating pitches grids is based on the famous [4]**Fibonacci** Series. In the Fibonacci sequence of numbers, each number is the sum of the previous two numbers, starting with 0 and 1. Thus the first numbers of the sequence are:

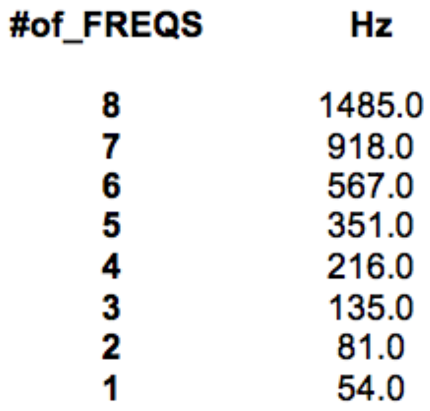
0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, …etc…

The ratio of two consecutive Fibonacci numbers numbers converges on the Golden Mean Ratio (approximately 1:1.618) as its limit.

The computer music composition **Stria** by **John Chowning** was based in almost every structural aspect on this ratio.
Stria uses this sequence of numbers for compile the grid pitches. In this panel are shown the few parameter you can use for the purpose to create a certain Fibonacci pitch grid. In doing this you have to specify:

Basel Frequency (fundamental frequency)
First of Series (the first member of the series)

[4] **Leonardo of Pisa** (c. 1170 – c. 1250), also known as Leonardo Pisano, Leonardo Fibonacci, or, most commonly, simply Fibonacci, was an Italian mathematician, considered by some "the most talented mathematician of the Middle Ages". (Wikipedia)



#of_FREQS	Hz
8	1485.0
7	918.0
6	567.0
5	351.0
4	216.0
3	135.0
2	81.0
1	54.0

You can obtain the *i-th* value of frequency using the next recursive formula:

$$f_i = f_0[\varphi^{(n+start)} - (1- \varphi)^{(n+start)}]/sqrt(5)$$

where: f_0=fund

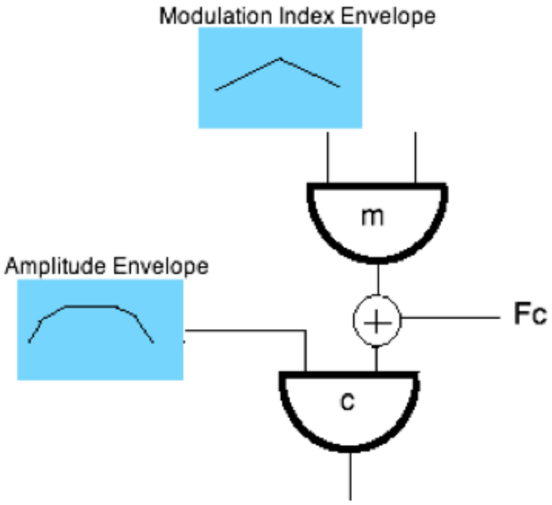
Scalar Spectral Compiler, distributes the number expressed in **Partial** between **Minimum/Maximum Frequency** Range.

Temperated take a input parameter for **Base Frequency**, expressed in MIDI notation and generate a chromatic temperate scale until Nyquist is reached. N.B. The MIDI note number 60, it means C3 (261.6 Hz)

FM Setup

As stated in the overview section, the main audio generator is capable of generating blast of notes whose base frequencies derive from the actual **Spectral Pitch Grid** (used only for Pad 1). At the basic level, the audio engine creates a simple band of tones centered on the actual value of the X pointer and distributed over a **Frequency Spread** (Event Parameters). Each tone consist of a cloud of notes controlled by a symmetrical and smoothed envelope. The length of each note is controlled by the parameter **Event Duration** (from 0.1 to 4.0 secs) while the rapidity of note scheduling is controlled by the parameter **Event Rate** (from 0.25 to 10 note for sec). Each note consists of an alternatively Frequency Modulation (**FM**) or simple wavetable (**WS**) based synthesis tones (only Pad 1).

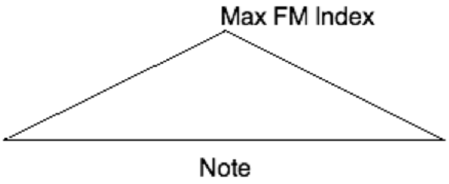
By selecting **FM** mode, the basic notes are generated by a classic couple of modulating/carrier oscillators controlled by independent amplitude and modulation index envelopes as shown in figure below:



The triangular shape of the modulation index envelope, creates a tone color change starting from pure sine at the very beginning, increasing to maximum complexity at the center, corresponding to the value of maximum **FM Modulation Index** (Event Parameters) and back to sine at the end of the envelope. The FM Modulation Index (Event Parameters) can vary from a minimum value (namely zero) up to a maximum of 4. This limitation is to prevent or reduce aliasing distortion when using high frequencies on the grid and or the maximum number of voices in the mixture.

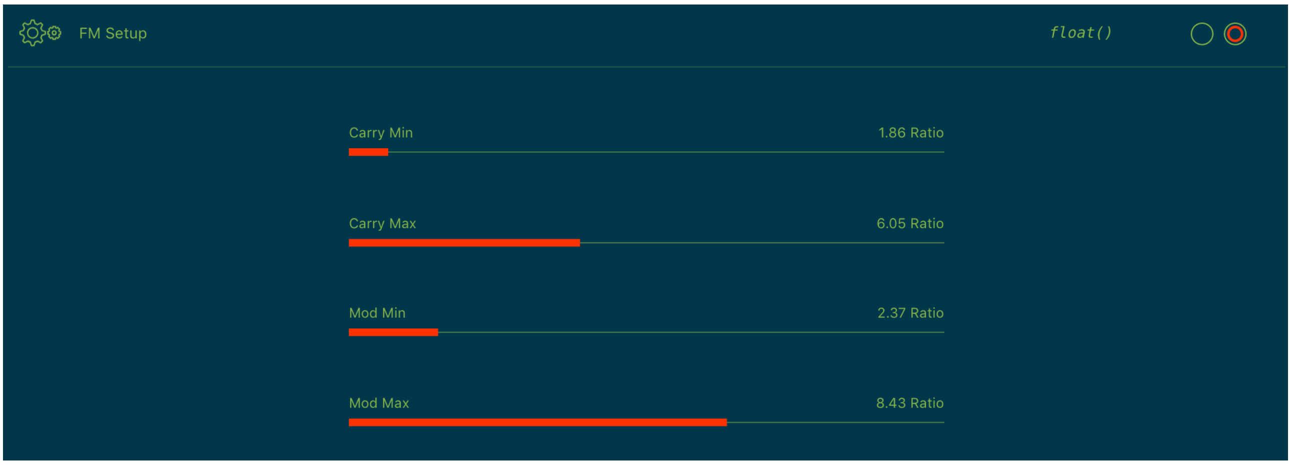
N.B. You can toggle Pad 1 in FM/WS, while the Pad 2 will work always in FM, therefore, it subject the FM Setup.
The Pad 2 is always affected by the from ‘FM Setup’ configuration, the amount of 'Car/Mod’, are rescaled from Y scrub.

Event Envelope, (Pad 1 and Pad 2)
You can vary the attack and release of the carrier amplitude envelope moving the Knob *Attack* and *Release*, both within the values 0.05 – 0.9 sec. During the course of one note generation, the modulation index always vary with triangular shape from zero to the actual selected value as shown in figure:



In the basic FM synthesis it is important to select the ratio of carrier to modulating ratio frequencies (carrier : modulating).

In the FM Setup you can see the controls for set this values. Note that you have not just two controls for this purpose but four. In fact in this implementation, the value of c:m is selected randomly inside the ranges you will specify.



Carry Min 1.86 Ratio

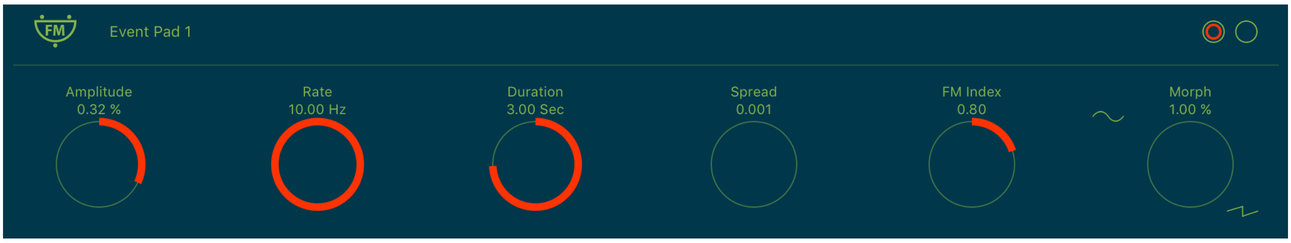
Carry Max 6.05 Ratio

Mod Min 2.37 Ratio

Mod Max 8.43 Ratio

In the case shown in figure above, the two sliders on the top are selecting the carrier frequency in the range 1.78 to 5.16 while the others two sliders are selecting the modulating frequency in the range 1.840 to 3.280. This values are thought as normalized frequency values so the actual values of frequency is obtained as a multiplication of this values with the frequencies of the pitch grid. Since the Selector on the right tools bar **Int/Floa**t is on the Int (Integer numbers) position, the program will select, on for each generated note, the integer part of random values for carrier and modulating frequency. So in this case, for the carrier frequency, the possible values are 1,2,3,4 or 5, while for the modulating frequency the possible values are 1, 2 or 3. For this reason the resulting spectrum is always at least harmonically related. On the contrary, when the **Int/Float** is on the Float position, the random values are selected as floats numbers inside the same interval. In this case the sound will be more inharmonic. You can switch from Float to Int whenever you want but the change will affect only the new generates notes so for a short portion of time the two types of spectra will cohabit and overlap.

Event Parameters Pad 1 (only for Pad 1)


Event Pad 1

Amplitude 0.32 %

Rate 10.00 Hz

Duration 3.00 Sec

Spread 0.001

FM Index 0.80

Morph 1.00 %

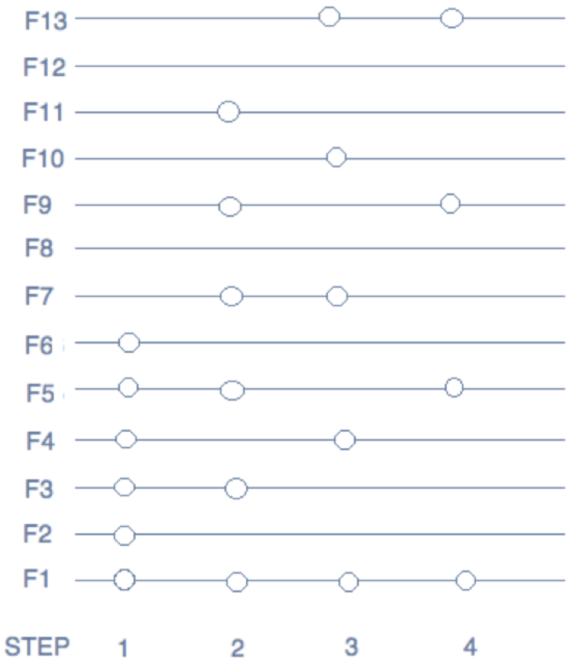
The farthest right selector of the ‘Event Pad 1’ is used to switch between complex (frequency modulation) and simple (wavetable) synthesis. In other words, checking the **FM/WS**, each note is generated by simply reading a pre-selected wavetable. Also in this case, only the new generated notes will be affected by the synthesis change.

Regarding of the **WS** sound generation, you can select among different wavetables built-in. There are eight basic waveforms available (see the Basic Waveform section code): the simplest is a sine wave while the others are more complex. The wavetables are associated with the Stria Parameter: **Wave Source Morph**.
Moving **Wave Source Morph**, you will obtain a new shape by a *linear combination* between built-in waves. When you move the slider the timbre changes smoothly between each table to each other. Use caution when selecting very complex waveforms to avoid possible aliasing effect.

Event Mixture (only for the Pad 1)

All the sounds produced in the manner described above, can be thought as “special kinds of notes” with a simple (WS) or complex (FM) timbre. In addition, the program creates some blurred note blasts that can be focused (or not) in frequency by superimposing a great number of overlapping notes with small random frequency deviations. If all this is not enough, *Stria* is able to create a “special kind of chords” by the superposition up to 8 notes (each one with the individual timbre and with the focusing attribute): this is what we would claim the *Mixture mode.*
The Mixture mode is automatically activated by setting the **Event Mixture Pad 1** (N.B. only for the Pad 1) at values greater than 1. When **Mixture Components** = 1 the chord degenerates in a single (even complex) note. The value set with this control determine the number of active voices in the chord so you can move the slider from 1 up to 8.
The basic note of the chord (a kind of chord root) is coincident with the frequency of the X control of the left most virtual **Pad** while the other notes of the chord are automatically derived from the actual selected pitch grid. So, for

example, if you have created a geometric grid based on the ratio 1.5 (perfect natural fifth) starting from 100 Hz as the lowest frequency of the pitch grid and you have selected 8 voices (the maximum value) for each chord, each chord will be made of 8 notes.
Moving the X **Pad** control the chord will change accordingly to the addressed values of the pre-computed pitch grid. It is important to notice that, if the X control is in a different position (i.e. in the position or X index = 2), the chord will have as root value the second frequency of the base grid. In other words, the chord root starts from the index value (X position) inside the pitch grid. The higher the chord root, the more bright and higher is the resulting sound quality.
In addition to the root chord, you can select its "density" specifying the distance between the chord notes via the **Mixture Grid Step** (Stria Parameters) and its relative slider. When the slider is at minimum level (**Mixture Grid Step** = 1): this means that each note in the chord is selected doing a unit step in the pitch grid. But, for example, if you select **Mixture Grid Step** = 2, the root note will be the same but the others are selected skipping one value, as is schematically illustrated in figure below. In it are represented four different notes placement depending on different values of parameter **Mixture Grid Step**. By varying the **Mixture Grid Step** value (from 1 to 9), you will change the distribution (or density) of the chord notes.



This effect produced by the incrementing of the STEP parameter is particularly interesting when using the harmonic grid. Notice that, with high values of STEP, one or more notes of the chord may fall outside the grid: in this case, the notes in excess are simply ignored.

Any change in frequency are always mapped on the actual grid, but since it will affect only the new notes, you will ear a kind of smooth sound timbre transformation rather than a simple change in pitch. In that sense Stria is more than any other thing a tool for creating spectra transformations. The response time on changes depends essentially from two parameters that control the attack and release time of the internal note envelope. In the right side of fig. 15 there are two sliders for that purpose:

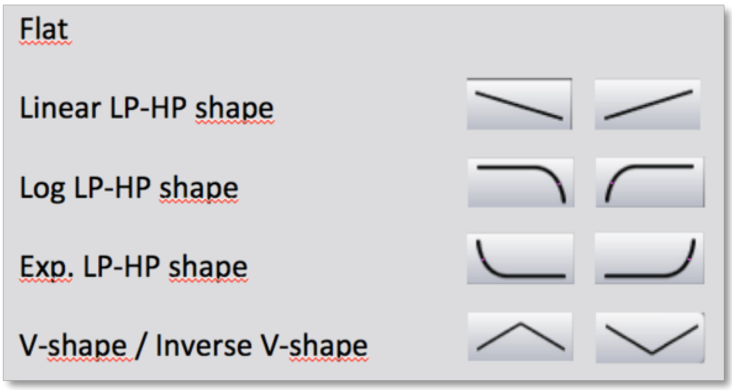
a) the attack time Event Envelope (Pad 1 and Pad 2) Attack / 2

b) the release time Event Envelope (Pad 1 and Pad 2) Release / 2

You will obtain smoother transformations when both the parameters are at maximum level but it is possible to achieve any desired combination of attack and release proportion.

Mixture Shape (only for the Pad 1)

The mixtures are mainly defined in the frequency domain but using this control it is possible to shape the amplitudes of the mixture components.
Rescale the mixture amplitude with selected mask. This behaves as a sort of spectral equalizer which work at mixture level. For instance, with a mixture of eight frequency (maximum possible), each components will amplitude rescaled according to selected mask. So, you can emphasize a part of mixture.
You have nine shapes that represent the “spectral envelope” of each mixture.



Flat
Linear LP-HP shape
Log LP-HP shape
Exp. LP-HP shape
V-shape / Inverse V-shape

Tips & Tricks

- Double tap on the **Knobs** to show the options. In the ‘**Edit**’ field you can enter the value from the keyboard. Tap ‘**Random**’ to generate a random value, ‘**Default**’ to reset to the factory settings and configure the knobs dragging behavior. Step fine resolution sets how many points of movement result in a one degree rotation in the knob's position. Only used in the horizontal/vertical slider modes. You can set Knobs behavior from Settings.

- Long Press Tap on the **Toggles** to open Control Manager Assignations.

- *Stria* can play audio in the background and/or mix it’s own audio with other iOS apps. Stria also supports Apple IAA (i.e. Inter-App Audio for iOS 7 or higher) and Audiobus, so you can send and receive audio through applications.

- Swipe (right to left) on Preset, Presets Bank or Samples Manager (non built-in file) to delete.

- You can load Presets (current selected bank) by remotely sending a MIDI Program Change message. The channel listening is the same for the MIDI Note On/Off, You can configure this setting through ‘Channel’ in the MIDI Manager (see later for further details). The available range is 0 ÷ 127. A value of zero will select the first preset in the list, a value of 1 for the second etc… But no bank selection is supported yet.

- We suggest you disable the screen rotation, otherwise the accelerometer doesn't work properly.

- The PAD 1 and PAD 2 Buttons, in the left area of each Pad (i.e. Pad 1 on the left and Pad 2 on the right), triggers on/off the note-event. When the button is selected, the note-event will continue even when you lift your finger from the Pad.

- **Init** will reset to default values and removes the Control Manager assignations, a reset message will be displayed.

- **Create Random Sound** values for all of the parameters, including the Effects. The result can be very interesting and wayward.

⏻ **MAINS Toggle** on/off DSP (Digital Signal Processing). *Stria* supports mixed audio with other applications, you can also disable DSP in order to save battery and CPU consumption. You cannot toggle off the Audio Engine when it’s connected to Audiobus or Inter-App Audio.

✥ Enable/Disable Accelerometer X/Y and LFO, see the Control Manager for more details.

● Captures Output stereo signal (post Effects) and creates an audio file *wav* format 44.1 kHz 16 bit in the Documents Folder. Files created by *Stria* are saved with an id-name followed by a number, for example:
Output Rec 1.wav, Output Rec 2.wav etc…
Each new file will be numbered with the next consecutive number.

🔊 Sets the iDevice hardware **volume**.

SYNC

🔗 The goal of SYNC is to setup a global Metronome (MASTER BPM). This will affect the LFO (Low Frequency Oscillator) and/or the Sequencer when its enabled to receive the SYNC signal - see Control Manager for more details. The SYNC can be controlled remotely by MIDI Clock signal, or via Ableton Link.
You can adjust the BPM (Beats Per Minute) through the minus/plus buttons and these will replace the current value with the integer BPM (previous/next) or you can TAP to set your own Tempo. When the MIDI Clock is activated (see MIDI Manager), the 'MASTER BPM' slider will be disabled and the Tempo value is controlled by the MIDI Clock.

Ableton Link description
Ableton Link is a new technology that synchronizes the beat, phase and tempo of Ableton Live and Link-enabled iOS apps over a wireless network. It lets you play devices together with the freedom of a live band. Anyone can start and stop their part while others keep playing, and anyone can adjust the tempo and the rest will follow. You can use Link to play with several copies of Ableton Live, with Live and iOS apps, or with iOS apps together – even without Live in your setup.

By default 120 BPM corresponds to 1 Hz frequency for the LFOs.

The ‘Sync’ button waits for the next beat (Tempo Forte) from the MIDI Clock or Ableton Link and resets all phases for LFOs, Sequencer etc.

You can enable the MIDI Clock Signal by following these steps:

1 Open MIDI Manager

2. Send a MIDI Clock Signal
3. Enable the MIDI Clock for the chosen Source, see MIDI Manager for further information

Samples

N.B. *Stria* does not use audio file for synthesis.
N.B. *Stria* does not have any Built-In audio files.
N.B. Samples Manager serves for only files sharing.

Samples manager can read/write in the Documents folder, you can also share files through AudioCopy, Document Picker (iCloud, Dropbox etc…) and AudioShare. Samples manager can also read the Built-in files, but you cannot rename or delete them. Touch on a file in the list to show the actions popup. Here it is possible to **Load** a file in one of two Samplers (i.e. Sampler Ch.1 and Sampler Ch.2), Open In... contains all system actions over the file, according to file type. For example you can send files as an e-mail, open file with a listed app etc…

✎ You can rename files in the Documents Folder, if you omit the file extension then Samples Manager will fill the original audio file extension for you. N.B. Samples manager shows only audio files supported, the other files in the Documents folder will not be seen.

▷ You can pre-listen to the samples before you load them.

🎙 Starts audio capture from the microphone immediately and saves new stereo audio file in wav format 44.1 kHz, 16 bit in the Documents Folder. Files created are composed by id-name followed by a number, for example:
Input Rec 4.wav, Input Rec 5.wav etc…
Every new file will be numbered with the next consecutive number.

N.B. You can record a mono or stereo file in wav format 16 bit 44.1 kHz according to the iDevice channels. Even if the Stria is connected to Audiobus or IAA, the Grab Audio will record always from iPad microphone/line

📥 You can import files from AudioCopy, Music Library, Document Picker (iCloud, Dropbox etc…) and AudioShare.

Access to the iPod library, this allows the application to read user’s songs, audio books, and audio podcasts. The file can be converted in *.wav* (PCM 16 bit 44.1 kHz) or *.caf* formats and stored in the local app Documents Folder. If the file exist at path then progressive number will be attached to the file name.

The document picker feature lets users select documents from outside your app’s sandbox. These include documents stored in iCloud Drive and documents provided by a third-party extension. Users can open these documents directly, editing them in place. This access simplifies sharing documents between apps and easily enables more complex workflows.

Add files through iTunes
• Connect your iPad to your computer and then open iTunes. Select your iPad under the DEVICES header in the left navigation column.
• Click on the “Apps” tab above the main window.
• Scroll down to the “File Sharing” section.
• Select *Stria* from the list of Apps.
• Under “*Stria* Documents” click the “Add” button; this opens a dialog box to select the location of the presets on your computer.

MIDI

You can enable/disable the incoming MIDI signal from all the physical, virtual and network ports available. As for the MIDI ports, you can enable/disable the MIDI Clock on a port and the configuration is saved as default and restored when you open the app.

If the Core MIDI Network is connected a wifi symbol will appear in the Network port cell.

Only ports that receiving MIDI Clock will be displayed as icons, touch the clock symbol (if present) to enable/disable receiving MIDI Clock. It will change its color from red to blue. You can receive MIDI Clock from one port at a time, you can enable/disable the clock for the current port or mutually selects another one in the list.

You can receive MIDI Clock from one port at a time, you can enable/disable the Clock for the current port or mutually selects another one in the list.

Channel sets the input channel for the selected port, the *iVCS3* DK (Keyboard) will receive the Note On/Off messages in OMNI (from all 16 channels) or, one of the 16 channels.

Configure Bluetooth LE MIDI

With the release of iOS 8 and OS X Yosemite, sending and receiving MIDI data is supported using Bluetooth Low Energy connections on any iOS device or Mac that has native Bluetooth Low Energy support. All established connections are secure which means that pairing is enforced and connections cannot be made to your devices without your explicit consent. After a connection is established, it simply appears as an ordinary MIDI device that any MIDI application can communicate with.

Enable NRPN

NRPN stands for "Non-Registered Parameter Number" and is part of the MIDI specification for the control of electronic musical instruments. NRPNs allow for manufacturer-specific or instrument-specific MIDI controllers that are not part of the basic MIDI standard.

Unlike other MIDI controllers (such as velocity, modulation, volume, etc.), NRPNs require more than one piece of controller data to be sent. First, controller 99 - NRPN Most Significant Byte (MSB) - followed by 98 - NRPN Least Significant Byte (LSB) sent as a pair specify the parameter to be changed. Controller 6 then sets the value of the parameter in question. Controller 38 may optionally then be sent as a fine adjustment to the value set by controller 6.

This fine adjustment is part of the conventional MIDI controller specification, where any of the first 32 controls can be optionally paired with a control offset 32 higher. This is the rare 14-bit Continuous Controller feature of the MIDI specification, and NRPNs simply take advantage of that existing option in the same way offering 16,384 possible values instead of only 128.

(http://en.wikipedia.org/wiki/NRPN)

When you enable NRPN, you should not employ CC 99 and 98 as the normal CC in order to avoid an undesired overlap. An NRPN message must be sent with this order: CC 99 98 6 38 where the first couple is the CC number and the second couple is the data. You can MIDI learn as for the normal CC, and NRPN works for all knobs and parameters. Although an NRPN CC number can be greater than 127, you will not be able to learn CC beyond 127.

Presets

Preset Manager will collect all the **Banks** *(i.e. .stria suffix)* and **Snapshots.** A *Snapshot* is a simple “photo” of the parameters in the UI (i.e. User Interface). Both the Matrix and Keyboard/Sequencer are included but they can also be excluded, see the Settings for more details.
So, once you find an interesting sound you can take a snapshot (i.e. save this new preset) by touching the ‘plus’ icon and this new preset will be added to the selected Bank. The top four elements of the Snapshots list are assigned for ‘HV Snapshot Pad’, you can then sort or delete the Snapshots in the List by tapping the ‘List Icon’. You can even share the selected Bank and you can restore the factory Built-In Banks from the ‘Cloud’ icon, there is two options:

1) ‘Overwrite’ will erase all user presets that have been saved in the factory preset-banks and give you the unaltered factory preset-banks back;
2) ‘Merge’ will preserve the preset-banks which contain the user presets; If you created your own user preset-banks and saved presets there they will stay untouched.

Snapshot banks are located in Documents folder, you can access files via iTunes Sharing (see Add files to Stria through iTunes Sharing).

↻ You can add new files by copying in the app's documents (through iTunes or from another app) and refresh the list from there.

🔀 Load a preset, randomly from the banks list.

🗑 Delete all Presets for a selected bank. A warning message will be displayed.

The Snapshots bank file is a kind of Property List format (*.stria*), you can edit as text and modify the values just rename in *.plist*

Swipe (right to left) on Snapshots or Snapshots Bank, to delete.

⊞ The purpose of the unique **Snapshot Pad** is to obtain intermediate values between four snapshots (corners). You can change snapshots assigned in the corner through the Snapshots list (see above).
The Snapshots Pad performs complex interpolations between four snapshots on a two-dimensional space. By scrubbing your finger on the Snapshots Pad

a temporary new snapshot will be created by mixing the four snapshots depending on distance. Thus you can modify hundreds of parameters with a single touch, this is also called *Hyper Vectorial Synthesis.*
Snapshots Pad can be resized on fullscreen for detailed controls (left bottom icon).

N.B. Snapshots Pad have no effect for Switches

Control

MIDI

Scrub Y

Accelerometer X

Accelerometer Y

LFO

After selecting **MIDI**, Stria enters into "learn mode" i.e. it “hears” incoming MIDI Control Change messages. It’s then possible to send messages with MIDI hardware, Virtual MIDI and Network MIDI. Flashing screen means the association between the parameter and the MIDI message has been successful and displays the corresponding control Change and Channel. Alternatively you can enter direct CC and MIDI Channel number through number keyboard. You will also receive an additional setting: 'Widget Range' that allows you to resize the range of action on the parameter. This is because you can use all 128 steps MIDI to control precisely particular part of the parameters.

Scrub Y binds the widget to vertical movement on the **Pad 1** (Scrub Y). An additional setting allows you to specify the speed (gravity) of the action. Please note that only Pad 1 will work for Scrub Y assignations.

n.b. Default Pad Y is designed for changing pulse duration, from top to bottom, you will decrease pulse duration.

Accelerometer X and **Y** are used to control the parameters through the horizontal or vertical inclination of the device.

LFO, Low Frequency Modulation is a table look-up oscillator with seven built-in shapes. The frequency of LFO is expressed in Hz and BPM (Beats per Minute). You can get the period time by dividing 1/freq. The LFO Frequency value is converted in BPM (Beats Per Minute), you can adjust this value through the minus/plus buttons, these will replace current value with the integer BPM (previous/next) or you can TAP to set your own Tempo.

ɸ Will reset to zero the phase of current LFO.
Φ Will reset to zero the phase of ALL LFO actives.

🔗 Enabling the Link symbol, you will disable the ‘Low Frequency Oscillator in Hz' Slider, so the value of the current LFO is sampled from the global ‘MASTER BPM’. You can adjust the BPM value from SYNC view. When the MIDI Clock or Ableton Link is enabled (see MIDI Manager and SYNC) the BPM value will affect all LFO enabled for SYNC mode. The goal of the ‘SYNC Subdivision’ Slider is to rescale the ‘MASTER BPM’ from SYNC according to the relative symbol (abbreviate musical note time values):

Normal : 1/1, 1/2, 1/4, 1/8, 1/16, 1/32, 1/64
Triplets : 1/1T, 1/2T, 1/4T, 1/8T, 1/16T, 1/32T, 1/64T
Dotted : 1/1D, 1/2D, 1/4D, 1/8D, 1/16D, 1/32D, 1/64D

Where ‘D’ is a **dotted** note and ’T’ is **triplet**.
With 1/4 (Default), the BPM follows the 'SYNC BPM’ or 'MIDI Clock’ BPM when enabled, i.e. 120 BPM corresponds to 1 Hz.

- 1/4 - each step is a quarter note (or BPM), e.g. 60 BPM is 0.5 Hz
- 1/4D - each step is a dotted quarter note (or 1.5 x BPM),
- 1/4T - each step is a triplet quarter note (or 0.5 / 3) etc.

♩ Reset default value (1/4 = 120 BPM)

Effects Chain

From left to right we have: Doppler Shift, Stereo Chopper, OOR (Out Of Range Glitch Generator), Synchronous Granulator, Spectral Arpeggiator, Reverb (8 delay line stereo FDN reverb, based on work by **Sean Costello**), Dynamic Compressor/expander.

The first effect of the chain is **Doppler Shift** which receives signal from the **Stria Engine**. If enabled, the processed signal will pass to the next effect (Stereo Chopper) or bypass if disabled. You will see enabled effects on the toolbar icons (orange color).

Stereo Chopper is a simple model of a helicopter/chopper simulation feed from *Stria* signal by **Oeyvind Brandtsegg** (1999).

A special Effect is **OOR** (i.e. Out Of Range) who is a glitch generator according to incoming threshold signals. An amplitude selector feed a second order resonant filter while scattering randomly the Band Width and Frequency cutoff.

Spectral Arpeggiator, arpeggiates spectral components, by amplifying one bin and attenuating all the others around it. With own LFO it will provide a spectral arpeggiator similar to Trevor Wishart's CDP program *specarp*.

Compress

Compress, limit, expand, duck or gate an audio signal.
This unit functions as an audio compressor, limiter, expander, or noise gate, using either soft-knee or hard-knee mapping, and with dynamically variable performance characteristics.
The 'Compress' first examines the incoming signal by performing envelope detection. This is directed by two control values '**Attack**' and '**Release**, defining the attack and release time constants (in seconds) of the detector. The detector rides the peaks (not the RMS) of the control signal. Typical values are .01 and .1.
The running envelope is next converted to decibels, then passed through a mapping function to determine what compresser action (if any) should be taken. The mapping function is defined by four decibel control values.

Threshold - sets the lowest decibel level that will be allowed through. Normally 0 or less, but if higher the threshold will begin removing low-level signal energy such as background noise.

Low Knee, High Knee - decibel break-points denoting where compression or expansion will begin. These set the boundaries of a soft-knee curve joining the low-amplitude 1:1 line and the higher-amplitude compression ratio line. Typical values are 48 and 60 db. If the two breakpoints are equal, a hard-knee (angled) map will result.

COMP/EXP - ratio of compression when the signal level is above the knee. The value 2 will advance the output just one decibel for every input gain of two; 3 will advance just one in three; 20 just one in twenty, etc. Inverse ratios will cause signal expansion: .5 gives two for one, .25 four for one, etc. The value 1 will result in no change.

The actions of compress will depend on the parameter settings given. A hard-knee compressor-limiter, for instance, is obtained from a near-zero attack time, equal-value break-points, and a very high ratio (say 100). A noise-gate plus expander is obtained from some positive threshold, and a fractional ratio above the knee. Each application will require some experimentation to find the best parameter settings.

XFADE & Master

XFADE controller is a simple slider, allows you to mix (equal power) the two sound generators (i.e. PAD 1 and PAD 2).
The **Master** controller adjusts the general Audio volume (expressed in deciBell). Please note that the multiplications falls after the Dynamics Compressor/Expander which is the last audo module in the chain (when enabled). High values can produce audio saturations.

Settings

Sampling Rate, for full-range audio, the recommended sampling rate is 44.1 kHz. Using a lower rate *Stria* will reduce the number of samples that has to calculate, this will facilitate your Device's burden, but it will also reduce the frequency range. If your device is struggling at 44.1 kHz, you should try a lower rate and/or increase Buffer Size.

Buffer Size set the Audio Device latency. When you run *Stria* from **Audiobus**, or **Inter-App** this value will be taken from **Audiobus** or **Inter-App Host**. Warning! When an app runs first and keeps running in the background, the Sampling Rate and Buffer Size will impose for the whole iOS while running.

Maximum Polyphony sets the number of active oscillators which is dependant on 'Event Parameter Rate', 'Event Parameter Duration' and 'Event Mixture Components'. A large numbers of Events for second with long Events length, will perform until 40 voices of polyphony. Furthermore every voice can perform up to eight FM (i.e. Mixture Components) modules. In total Stria can perform up to 320 FM module (i.e. 640 wavetable oscillators). For old devices devices such iPad 2 or iPhone 4, It recommends a number of 10. Decrease the polyphony if you hear sound stuttering or increases to achieve best sound quality in the modern devices.

Snapshots Pad Resolution Update sets the time interval for the interpolations, from about 15 to 350 milliseconds. Short times for higher resolution but higher consumption of CPU.

Save/Restore the Last App Setup, when you exit from the application, save the status and recharge to the next opening.

Others Settings

The switch in this section are self-explanatory. As for all warning messages; include/exclude Control Manager, Assignations and Knob Color in the presets etc…

Knobs Interactions Mode

Use this to configure the Knobs dragging behavior. Step fine resolution set's how many points of movement result in a one degree rotation in the knob's position. Only used in the horizontal/vertical slider modes.