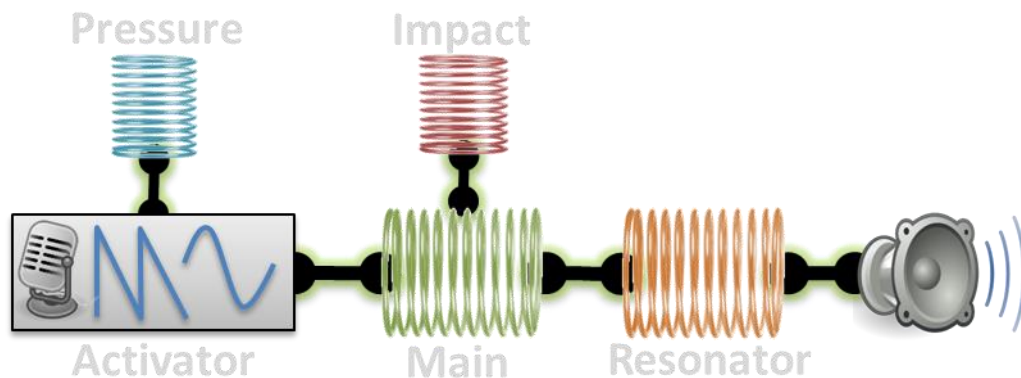




SpringSound



Overview

SpringSound is a physical-modeling audio synthesizer that produces sounds by simulating interactions and vibrations of a set of activators, and masses connected to springs. The model offers detailed control of an array of features including: gravity, bouncing, fluid and static friction, mechanical limits, non-linear functions, and more.

It can be used as a sound generator as an effect using its Audio-In feature.

SpringSound can be controlled by a standard MIDI keyboard and directly from the iOS device screen. With the exception of a reverb component, the audio output is generated entirely by the physical models. **SpringSound** is Audiobus® compatible, so you can route its output to external effects, recording and looping apps, and you can also use it as an effect entering a totally new world of physical-modelling effects.

SpringSound app is designed for optimum use with the Haken Continuum (www.hakenaudio.com) and integrate an IOS Continuum Remote allowing a very comprehensive and accurate remote control of the Continuum from you iPad (access to all parameters except programming the Eagan Matrix).

SpringSound has been designed to explore sound domains you won't be able to access with other synthesis techniques. It expands your sonic palette with a new world of physical sounds. It is a rich musical instrument, a sophisticated sound-design tool and an entertaining sonic playground.

SpringSound is the big brother of my former **Ackorage-Spring**



Main Features

- A unique, powerful, and fully editable, physical modelling synthesis engine,
- A refreshing alternative to common subtractive analog synthesis techniques,
- Detailed edit and real-time MIDI control of real-world parameters like Mass, Force, Gravity, etc.
- Can be used as a sound generator or as an effect processor,
- Create a wide range of rich sounds from realistic to other-worldly, from simple to *dangerous*,
- 88,200 Hz internal sampling rate,
- IOS Haken Continuum Remote (only on the iPad version),
- Core MIDI compliant (including virtual and network MIDI),
- Playable by any MIDI device,
- Playable via touchscreen with innovative physical controls,
- Effective preset import/export via iTunes File Sharing,
- Fully Audiobus® compliant including state saving,
- Inter-app audio compliant,
- Available on iPad and iPhone, with portrait and landscape mode supported.

First screen



The initial launch screen provides access to:

- The **Spring Engine**: the main play and edit screen of the physical model.
- The **Continuum Remote**: access to all Haken Continuum controls from your iPad.
- The **Configuration**: global settings (polyphony, audio buffers, etc.).
- The **Documentation**: in-app access to this user manual.
- The **Credits**: learn about the people and online resources that helped in the development of this app.

Main Function Buttons



Along the top of the Spring Engine screen are the main function buttons. Below the main function buttons, the Preset name is displayed. From left to right the main function buttons are:



Back to Main Screen

Go back to the start screen of the application.

Panic

Stops sounds and reinitializes the physical model. With a physical model, it is possible to create settings that cause the model to “break” (for example, a high force on very small mass, with no call-back spring force and no friction) resulting in out of control sound or no sound at all. This Panic button can be very helpful while designing and troubleshooting a new sound.

Hold

*The Hold button acts as a sustain pedal. Once sustained you can still interact with the sound by touching the screen and attracting the moving balls, or by tweaking the knobs. This feature is especially useful when using **SpringSound** as an effect processor.*

Freeze

This button allows you to pause (freeze) all the balls, interrupting all collisions. This is useful if you only want sounds from MIDI notes or screen touches.

Presets

*This icon provides an access to the Preset List (see **Presets** section below)*



Three other buttons are available along the bottom the screen:



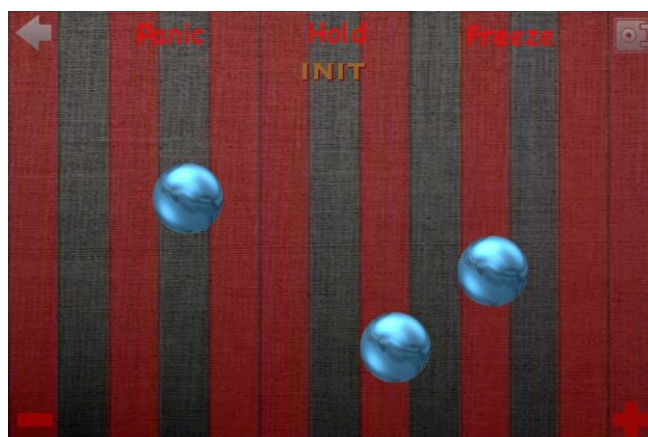
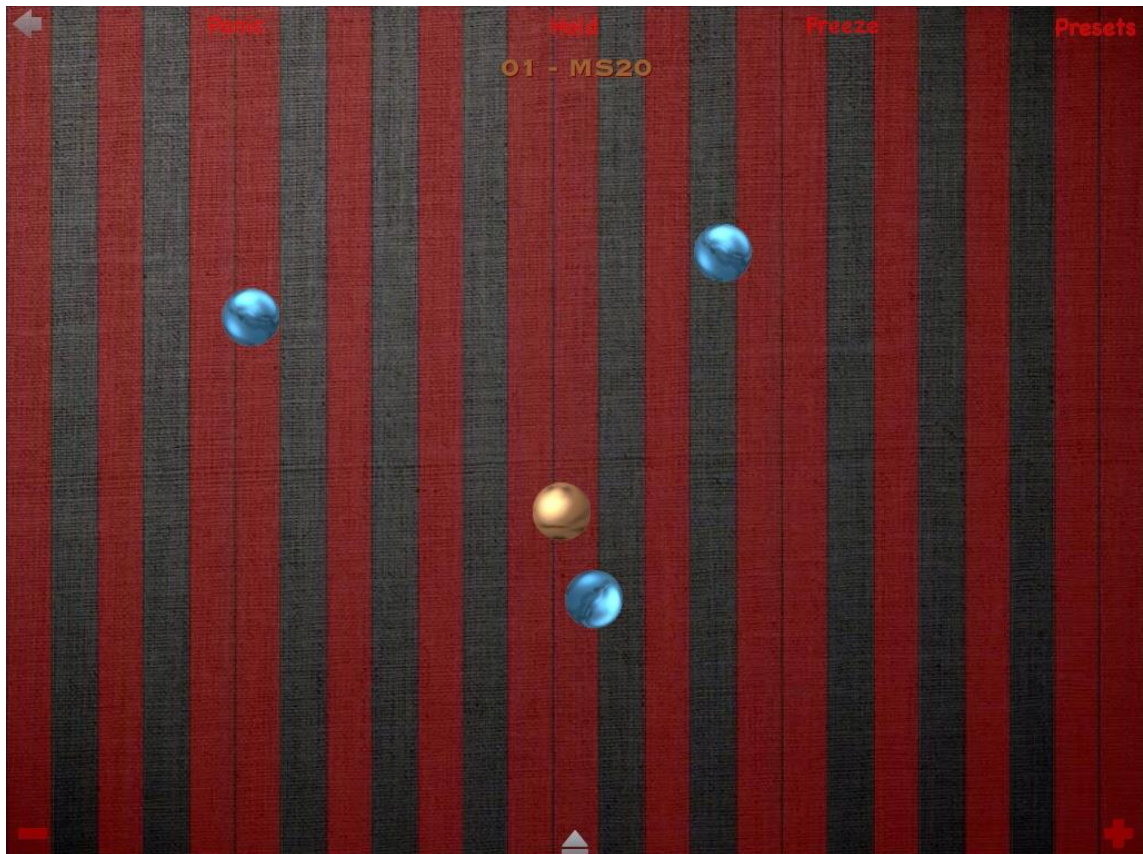
These two buttons allow to change the octave of the playing surface. You can go up or down two octaves around the middle position.



This button is the access point to all the physical model parameters. Once the parameters displayed, the button will reverse (upside down) and be available to hide the parameters. Note that the Edit screen is not available when on iPhone landscape mode.

On-Screen Instrument Control and Balls

The Spring Engine screen is the play-surface where the moving balls are displayed. It also provides access to the main functions, the play area, and edit section drawer at the bottom. The images below show the Spring Engine screen on iPad and iPhone.



There are two different types of balls: Note Balls and Active Balls:

Note Balls (the blue ones) are created when you play a note (with a Haken Continuum, MIDI device or touching the screen). Note Balls engage the 'spring engine' at the appropriate pitch. This means that when the ball will collide with another ball or with the edges, it will result an impact on the 'spring engine' (and usually some sound consequently, depending on the engine parameters). Depending on the chosen parameters the ball position can dynamically control some physical parameters or the pitch of the note. These balls are blue.

Active Balls (the cooper ones) are activated from Configure screen (you can have 0, 1 or 2 of them). They have their own 'spring engine' associated with them (but they can be "stolen" in case of low polyphony) meaning that there will be an impact on the 'spring engine' for each collision of the ball. These balls are self-animated and have the ability to move individually in a random manner.

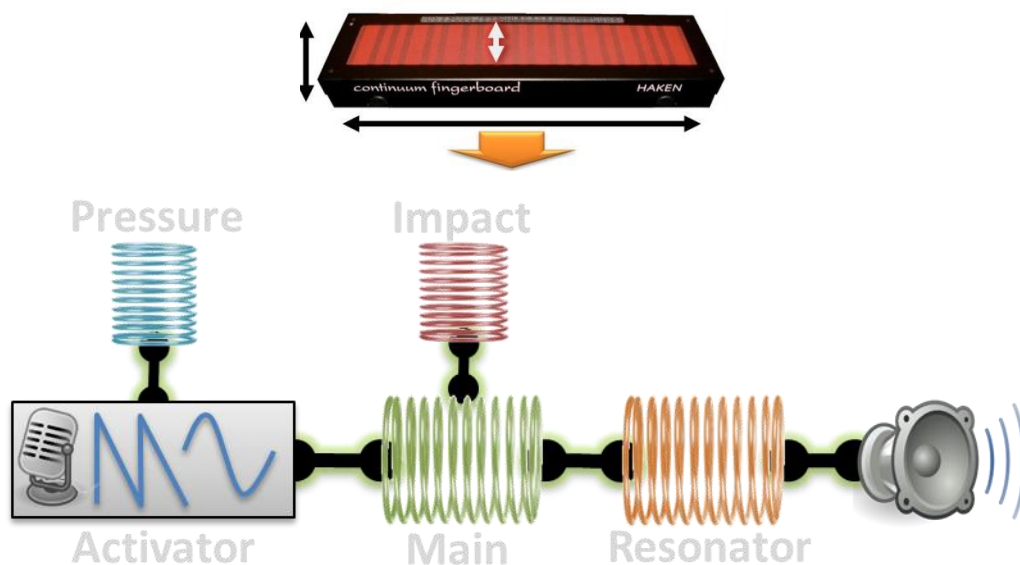
All types of balls are sensitive to gravity and therefore will fall on one side of the screen or the other depending on the angle of your device.

When you touch the screen a note is started. The pitch of the note depends on the left-right position of the touch. The vertical position of the screen touch is controlling both the Y and Z dimensions together (see below), which works as note velocity. Once a note started, you can move (drag) your finger on the screen and both pitch (X axis) and the parameters associated with Y & Z axes will respond in real-time accordingly.

On screen touch, the new blue ball is sent up with a force depending on the vertical position (velocity), and the balls will be attracted to your finger (each ball is attracted by the finger that launched it). In Hold mode (sustain) a touch on the screen attract all the balls together. When balls are fired by an external MIDI device, the behaviour is the same as in Hold mode.

Edit

At the bottom of the screen there is a small up-arrow symbol. When clicking on it, the model parameter panel will appear. (The panel can be closed/hidden by clicking the down-arrow).



Components of the Model

The above figure is an overview of the full 'spring engine'. It consists of: a control input source (i.e., the Haken Continuum, MIDI controller, or touchscreen input), four springs, an activator, a reverb (not shown in diagram) and the output.

MIDI input

SpringSound is designed to be controlled by any typical MIDI controller, and is also able to take full advantage of the Haken Continuum. It manages Note On and Note Off on any MIDI channel. It also manages two sets of MIDI controllers:

- MIDI controller 1 (modulation wheel), 2 or 74 (corresponding to the Continuum Y axis), and can be associated with several **SpringSound** parameter (depending on the parameters),
- 7 or 11 (and Aftertouch depending on the configuration), corresponding to the Continuum Z axis (pressure), and mainly controlling the activator and springs output level in **SpringSound**

On a usual MIDI keyboard, the Y group can be controlled by the modulation wheel. The Z can be controlled by a programmable control (like a knob or a fader) associated to control 7, 11 or by Aftertouch.

On the Configure page, the pitch bend range can be set to fit the Haken Continuum 48 or 96 half steps range, and additional 2, 4 and 12 semitones.

Still on this Configure page, it is recommended to use the "Continuum Mode" parameter when you control **SpringSound** from a regular MIDI device.

A special feature of **SpringSound** for managing MIDI polyphony is that when you replay the same note several times it will reuse the same 'spring engine' process (a bit like a piano re-striking same strings for a single note).

SpringSound is Apple Core MIDI compliant and can be used with any Core MIDI app (like LittleMIDI from Synthetic Bits or GeoSynth from Jordan Rudess's Wisdom Music) and any Core MIDI compliant interface (iRig, iConnect, Mobilizer II, etc.). It can also be controlled using Network (Wifi) and Virtual MIDI.

SpringSound provides control of all its parameters via MIDI CC's (continuous controllers). This gives you real-time control of all settings using the physical knobs of an external controller. Controls are detailed in the following table (starting from base control is 20, on a configurable MIDI channel, the default channel being 11).

Control Base 20+	Associated parameter
Spring parameters	
0	Mass
1	Gravity
2	Tune
3	Spring
4	Spring linearity
5	Limit
6	Friction
7	Friction linearity
8	Bouncing
9	Static friction
10	Dynamic friction
11	Threshold (friction)
12	Spring page Up or Down
Level parameters	
13	Global level
14	Feedback
15	Feedback delay
16	Feedback Modulation
17	Velocity Impact
18	Collision
19	Pressure
Modulation parameters	
20	Y Main Friction
21	Y Main Mass
22	Y Main Spring
23	Y Resonator Friction
24	Y Resonator Mass
25	Y Resonator Spring
Miscellaneous model parameters	
26	Coupling Level
27	Activator Stability
28	Select Activator
Reverb	
29	Reverb Level (dry-wet)
30	Reverb Type
Led toggles (routing)	
31	Impact to resonator
32	Tune resonator
33	Tune Impact & Pressure
34	Activator Tuning
35	Tune Delay
36	Damp on release
37	

37	Pressure ctrl on spring output
38	Damp Motion
39	Motion to Modulation
40	Motion to Pitch

Moving and colliding balls

The balls collisions will result in an impulse on the impact spring (a bit like kicking the spring). The harder the collision is, the higher the resulting impulse will be. The Collision parameter controls the maximum force value that will be applied on the impact spring.

Impact spring

The Impact spring is activated by the ball collision and by the note on. In case of note on, the velocity will set the level of the impulse. The output of the Impact Spring is fed in the input of the main spring (as an input force, or as a target position). Alternatively, the output of the Impact Spring can be used to tune the resonator.

The Impact spring can also be used to modulate the activator level.

Pressure spring

The Pressure Spring is designed to be controlled by the continuum Z axis, but is also controlled by the note-on velocity and external 7 or 11 MIDI controllers. The input of this spring is used as a target position for the spring centre position. Its output will control the Activator level. This Pressure Spring can be seen as a kind of ADSR envelope on the Activator level, but with a much more physical behaviour (note that this behaviour is very dependent on your playing style, reacting differently on slow or fast input change, and also having behaviour depending on the previous actions you had). The Pressure Spring can also control the output level of each other spring (except the pressure spring itself, of course).

Main spring

The Main Spring tuning is set by note played on the Continuum, the MIDI device or the screen touch position. It has the Impact Spring and the Activator as input. Its output is fed into the Resonator Spring.

Resonator spring

The Resonator Spring often has a fixed pitch, but can alternatively be tuned to the note. The connection between the Main spring and the Resonator can be either physical (the Resonator steals some energy to the Main spring) either theoretical (the Main spring doesn't lose any energy in the transfer). The output of the Resonator spring directly provides the sound of the 'Spring Engine'.

Activator

The Activator can be seen as a kind of magnetic field applying a force affecting the motion of the Main Spring. It can be a waveform (square, pulse, triangle or saw), an audio input (from the device microphone or from another app), a random field, or a static value (meaning that it will pass through the output of the Pressure Spring who is controlling its level). If no activator is selected, this also means that the Pressure Spring is ineffective. The Activator may be attributed an certain amount of instability.

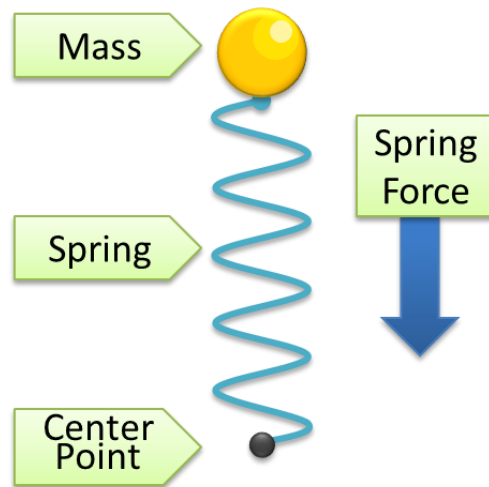
If you are not used to physical modelling, but have knowledge of more usual subtractive synthesis, keep in mind the following comparison:

- The Activator and Main Spring are together a kind of sophisticated oscillator (VCO);
- The Resonator Spring can be seen as a special kind of filter (or resonance chamber);
- The Pressure Spring can be seen as a very special kind of ADSR envelope (mainly on a VCA);
- The Impact Spring has no subtractive synthesis counterpart, but can be understood as a piano hammer, a plectrum, a drum stick, etc.

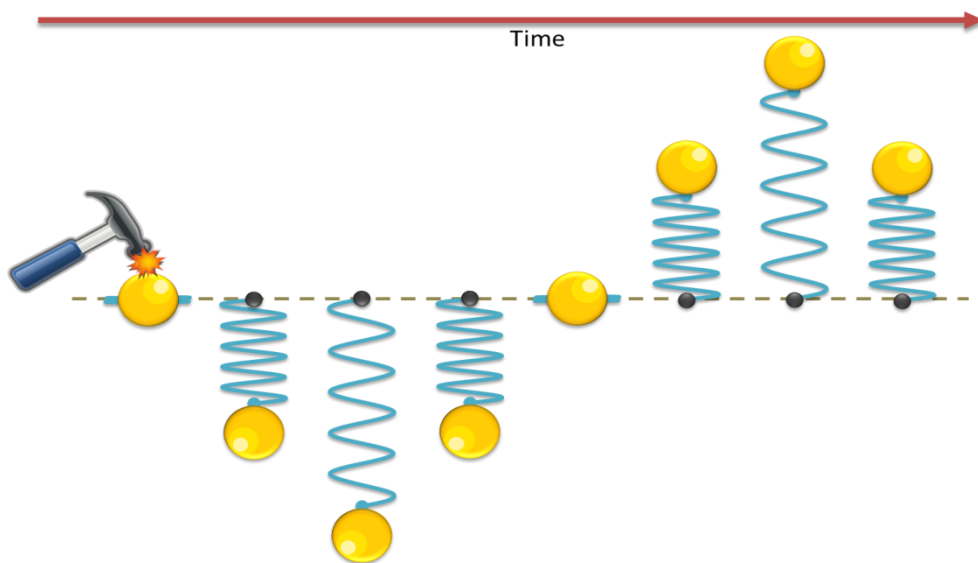
Of course, all this is just a basic analogy, and there are a lot of situation where the Spring model doesn't behave at all as any well-known synthesis style.

The spring

The spring model is the basic component of a 'spring engine'. It is basically corresponding to a weighted mobile body connected to an end of a spring, the other end of the spring being connected to a fixed point.



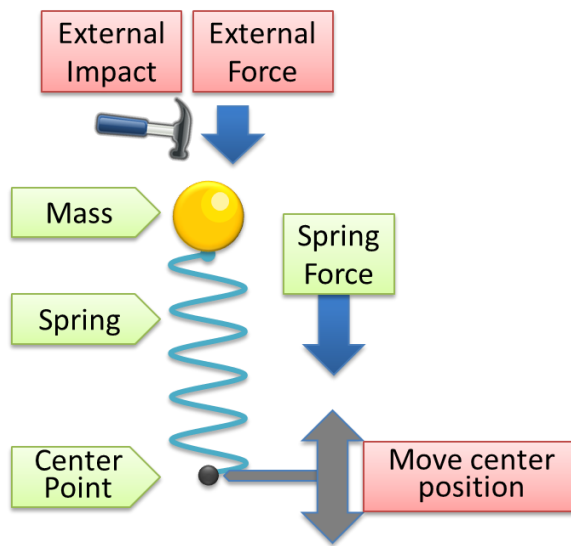
If some impulse (the hammer in the following figure) is provided to the spring at centre position, it will start to oscillate and this oscillation can be used as an audio signal. As there usually is some friction (fluid friction), the oscillation will progress lower and disappear. Note that the fluid friction is dependent on the speed (the higher the speed is, the higher the friction will be), and is acting as a damper. Another important fact is the more massive the weighted body will be, the longer it will oscillate (due to its inertia).



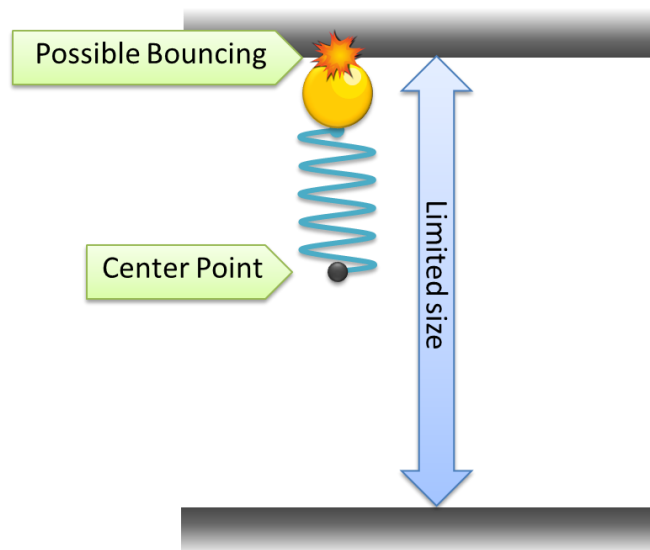
At this level **SpringSound** allows you to control the force of the spring, the linearity of this force, the mass of the mobile body, the fluid friction and the linearity of this friction.

With **SpringSound** it is possible to act in different ways on this mass-spring model:

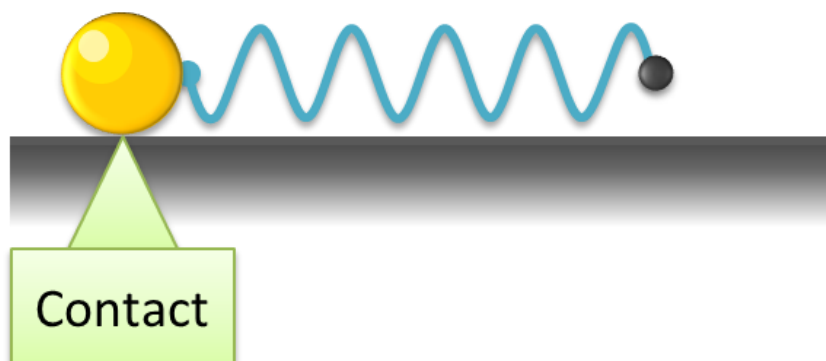
1. An impulse (a "kick") can be applied on the spring: a force is applied during a very short moment).
2. A continuous force can be applied on the spring (for example as if you were continuously blowing on it)
3. The centre position can be moved (so the spring will try to join this new centre position)
4. Gravity can be applied on the mobile body



Another possibility is to set a mechanical limit designating the maximum strain of the spring. It can alternatively define the upper and lower walls on which bouncing will possibly occur. *SpringSound* allows you to control the maximum size of the spring and the bouncing efficiency.



It is also possible to have a "dry friction". You have to imagine what occurs when sliding on the floor. In this situation there is a force that needs to be overcome to enable relative motion of mobile body (also called stiction). Unless this force is sufficient, the mobile body will stick on the ground. When it starts moving, another (smaller) force is slowing it due to the contact with the ground: the kinetic or dynamic friction. When the speed comes under a certain level, it will stick again on the ground. This is often called Stick-Slip mechanism. Note that the higher the mass is, the more it will stick on the ground. *SpringSound* allows you to control all the Stick-Slip parameters.



Spring model parameters

Each spring can be controlled through 12 main parameters.



Mass: Define the Mass of the mobile body connected to the spring. The mass can't be null. The higher the mass is, the more difficult it is to start moving, but the longer it will move, due to inertia. High mass also usually results in a bass sound.

Spring: Define the strength of the spring, the higher this strength is, the higher frequency you will get. The strength can be null: in this case nothing calls the spring back to its centre position (note that the centre position can be modulated, depending on the type of input you select for the spring).

Spring Linearity: Define the linearity of the spring. When set to 0 (centre position) the spring force is linear ($F=kx$, with k fixed, whatever the distance x from centre position is). If it is positive, the spring force will be lower when distance from the centre position is small. If it is negative, the spring force will be lower when distance from the centre position is high.

Friction: Define the fluid friction. This friction is depending on the speed. The higher the friction is the faster the spring will stop moving. Note also that the spring will continue moving if the mass is higher (for the same friction), due to inertia.

Friction Linearity: Linearity of the fluid friction. When set to 0 (centre position) the friction is linear. If it is positive, the friction will be lower when the speed is small. If it is negative, the friction will be lower when the speed is high.

Static Friction: Define the static friction. This sets a minimum force to be applied before the spring starts moving. Note that for high values of this parameter, the spring will often not be able to move any more. Also note that for higher mass, a greater force will be needed to start the motion.

Dynamic Friction: Define the dynamic friction. This is the friction that will be applied to the motion after it has escaped from the static friction. This friction is fixed (not depending on the speed as the fluid friction). The value is 0-1 expressing a portion of the static friction. 0 means no dynamic friction, and 1 means that the dynamic friction has the same value as the static one (it can't be higher).

Threshold: After the spring has escaped from the static friction, when its speed is reduced to a certain value (threshold) it will stick (stop) again. The Threshold parameter sets this minimal speed. The threshold can be 0 meaning that the spring will need to freely stop before sticking again.

Size Limit : Define a mechanical limit due to the size limit of the spring. If there is no bouncing, the spring motion will be quickly stop when reaching the limit. At this point it will be stopped, losing all its kinetic energy: so mechanical size limit is very different from a usual audio signal clipping. When set to 0 there is no size limit, meaning that the spring can move as far as it can from the centre position. Note that in such a situation, this can result in a very high audio level, which is limited by a soft clipping at the last stage of **SpringSound**, just before the audio out. When set to the maximum value, the spring only has a very tiny space inside which it can move. The led in the centre of the button indicates that a Size Limit is set (either by the parameter itself, either by the Bouncing parameter that requires some limit).

Bouncing : Define the bouncing efficiency. When set to 0, the spring loses all its energy when touching the edges. When set to maximum value, the spring bounces and keeps all its energy when touching the edges. Intermediate values sets the percentage of energy kept when bouncing. If Bouncing is set with no Size Limit, an arbitrary high limit is set (and turns on the led in the middle of the Size Limit button)

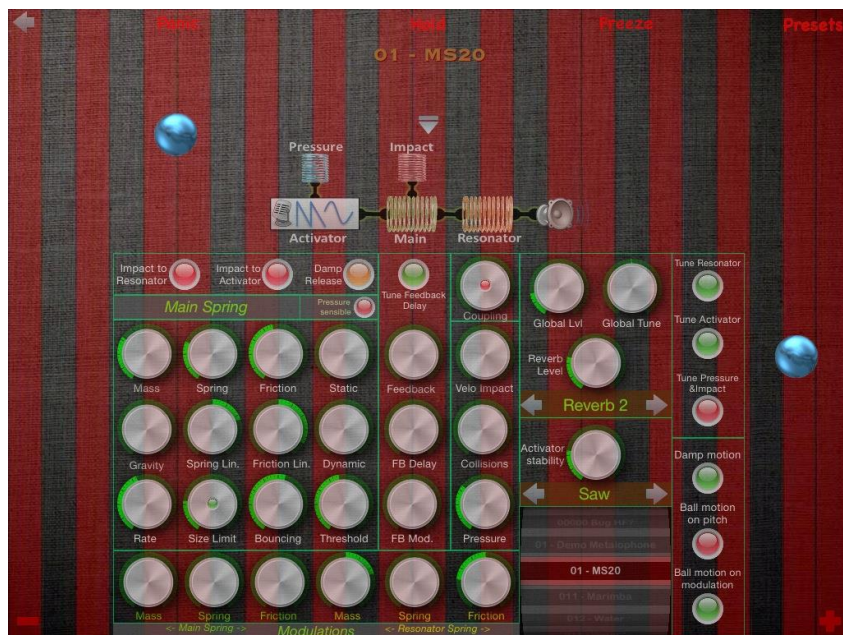
Gravity : Define the gravity level. Gravity is acceleration, and is not dependent on the mass (explaining why Elephant and Feather are falling at the same speed). Note that a high gravity may stick the weighted body connected to the spring at the low Size Limit (or somewhere very far away from the centre position if no limit is set), therefore preventing any motion (and any sound...).

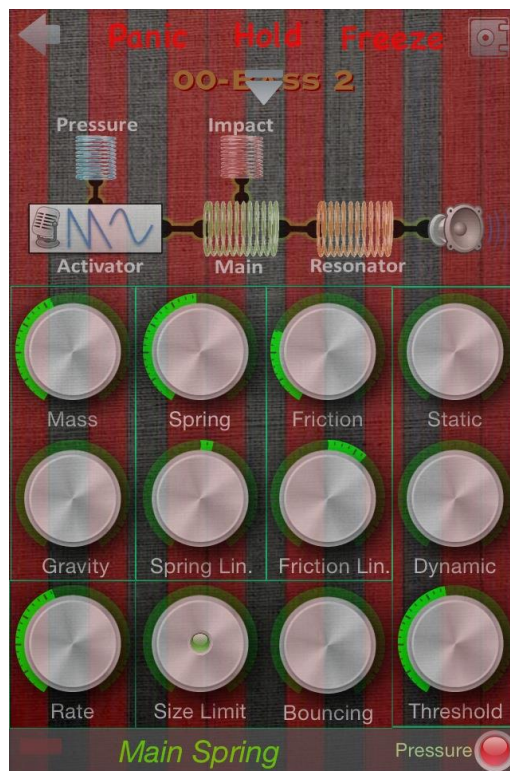
Rate (Tune) : The Tune parameter sets a time processing parameter of the model, and is useful to tune the spring.

Pressure sensible : At the right of the spring name a toggle (on-off LED) named "Pressure" controls the output of the spring. This control is driven from the Pressure Spring (so you don't have this parameter on Pressure Spring itself). When active, think of properly setting the Pressure Spring parameters and raise up the Pressure Level (see later on) in order to get sufficient output level.

To edit the parameters of each spring, just tap on this image of the required spring on the synoptic image, or tap on the spring name itself. The buttons will take the colour of the edited spring (green, yellow, red or blue) and the spring name will be updated.

The three images below show the full editing screen in landscape and portrait mode on iPad, and on iPhone (portrait only for edition).





29 additional parameters are available to control the 'spring engine'. The above figures provide an overview of parameters on iPad and iPhone.

Activator Type

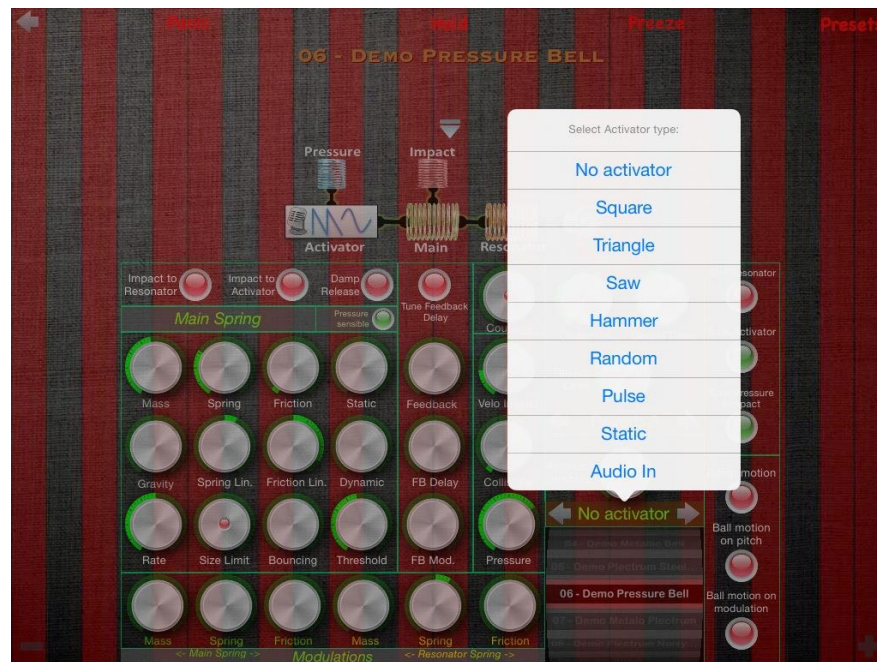
The activator is used to stimulate the main spring. It can be seen as a kind of magnetic field applying a force on the Main Spring. It can alternatively be used to provide a modulated target position (see "*Main Spring Input as Force or position*" parameter). The activator frequency is usually set by the note (Continuum X axis or keyboard or MIDI note), but can alternatively be a low frequency set by the Continuum Y axis or the MIDI Modulation controller (see "*Tune Activator*" parameter).

The output level of the activator is controlled by the Pressure Spring and optionally by the Impact spring. The possible values for the activator are:

Audio-in	Audio input from the device microphone or from another app (through audiobus).	
Square	a square wave field	
Pulse	a pulse wave field	
Triangle	a triangle wave field	
Saw	a saw tooth wave field	
Random	a random field	
Static	in this case, the activator outputs a static value (of 1), modulated by the output of the Pressure Spring. This means it passes through the Pressure Spring	
No Activator	no activation of the main spring	

Note that all waves are anti-aliased.

Activator Stability: Sets the frequency (in fact phase) stability of the activator. At minimum value, the Activator is totally stable, but a small instability is very often much nicer. Stability control is ineffective when the activator is Audio-In, Random, Static or No-Activator.



Control parameters

Global Level: This is the main output level. A soft clipping will occur in case of high level audio output.

Global Tune: This is the main tuning: tune all springs and activator together.

Feedback: The output of the Resonator Spring can be fed as an input force for the Main Spring. This parameter controls the amount of feedback. The parameter has to be used with care as it can lead to continuous high level sound due to Larsen effect. Note that the feedback steals some energy to the resonator spring therefore, depending on your spring settings, high feedback can also nearly mute the output.

Feedback Delay: The feedback may not be instantaneous (physically the signal may need time to go through the body and reflect, as happens for a string). The Feedback Delay can be set from 0 to 0.5 second. A additional toggle (see below) tunes the delay to the played note (therefore allowing control of phase issues, and corresponding to different sizes for instrument body depending on the notes, like a set of bells). If Tuned Feedback is selected the Feedback Delay is no longer a fixed value, and will only set a longer or shorter delay, always depending on the played note.

Feedback Modulation (FB Mod.): This controls the **Feedback** modulation from the Continuum Y parameter, MIDI Modulation controller or screen vertical position. When positive, the **Feedback** will increase for modulation values over 64 (centre position), and decrease below; when negative, the **Feedback** will decrease for modulation values over 64, and increase below.

Velocity Impact: The Velocity Impact sets the level of the note velocity (from the MIDI note, or the vertical position of a touch on the screen) that will be used as an impulse on the Main Spring when beginning on new note. This impulse only occurs once at the start of the note. A value of 0 means that no impulse will be applied on main spring, a value of 1 means a maximum level impact.

Collisions: The Collision parameter sets the level of the force of a collision (between 2 balls, a ball and the edge of the screen, or the impact on the ball created for every new note) that will be used as an impulse on the Impact Spring. This impulse occurs for each collision. A value of 0 means that no impulse will be applied on the Impact Spring, a value of 1 means a maximum level impact.

Pressure: This basically sets the Activator (including Audio-In) output level. The Pressure parameter sets the level of the Z Continuum axis (or MIDI controller 7 or 11, or vertical position on the device screen, or fixed by note on velocity on regular keyboard) that will be used to modulate the input of the Pressure Spring (which is

itself controlling the output of the activator). This input to the Pressure Spring is a target position, meaning that the output position of the spring will join the pressure level according to its parameters (for example: high mass and friction plus low spring force will result on a longer time to join the target position, lowering the friction will result on possible oscillations, etc.). A value of 0 means a target of 0 (so a 0 output), high values can be used to play with spring Size Limits. Remember to increase the value of this parameter if you use the Pressure toggle control on Springs (and also to control the activator levels).

Coupling: The connection between the Main spring and the Resonator can be either physical (the Resonator steals some energy to the Main spring) either theoretical (the Main spring doesn't lose any energy in the transfer). When you set Coupling to minimum, the centre led turn red indicating a theoretical coupling. Non null values define the amount of energy that will be stolen from the Main spring to activate the resonator. Note that in this kind of coupling, both spring have consequences on the global tuning (without activator, only the main spring sets the global tuning when theoretical coupling is chosen).

Y or Modulation parameters

All the modulation from the Continuum Y parameter (or MIDI Modulation wheel controller, also corresponding to a touch vertical position) behaves in the same manner. When positive, the modulation will increase from a minimum value (proportional to the modulation level) up to the set value of the parameter. When negative, the modulation will decrease from the set value of the parameter down to a minimum value (proportional to the modulation level).

Main Friction: sets the level of modulation of the Main Spring friction.

Main Spring Force: sets the level of modulation of the Main Spring force.

Main Mass: sets the level of modulation of the Main Spring mass.

Resonator Friction: sets the level of modulation of the Resonator Spring friction.

Resonator Spring Force: sets the level of modulation of the Resonator Spring force.

Resonator Mass: sets the level of modulation of the Resonator Spring mass.

Routing parameters and toggles

A set of 8 On-Off buttons (led lights) allow to control the routing inside the 'spring engine'.

Impact to Resonator: When set, the Impact Spring output is used to tune the Resonator Spring instead of kicking the Main Spring.

Impact to Activator: When set, the Impact Spring output is used to modulate the Activator level.

Tune Resonator: The Resonator Spring is often un-tuned (it has a fixed tuning), but if is this parameter is set, the Resonator Spring will be tuned relatively to the current note (it won't have the tune of the note, but will be updated proportionally).

Tune Pressure and Impact: The Pressure and Impact Spring are often un-tuned (they have a fixed tuning), but is this parameter in set, the Pressure and Impact Spring will be tuned relatively to the current note (they won't have the tune of the note, but will be updated proportionally).

Tune Activator: The Activator Spring is often un-tuned (it has a fixed tuning), but is this parameter in set, the Activator Spring will be tuned relatively to the current note (it won't have the tuning of the note, but will be updated proportionally).

Tune Feedback Delay: as explained for the Feedback Delay parameter, the Feedback Delay can be tuned. This toggle allows for a selection between tuned and un-tuned delay.

Damp: When Damp is selected, at the release of each note, the friction of Main, Resonator and Pressure Spring is raised up (in order to shorten the sound release). Two level of damping are available: smooth (orange led) or hard (green led).

Motion Damp: When set, the ball motion suffers a higher friction. It will therefore stop more rapidly and allow to keep each ball closer to its controlling finger (when played on the device screen). This can be especially useful when combined with **Ball Motion on Pitch** (to keep a usable tuning).

Ball Motion on Modulation: When set, the ball vertical position is used to control the modulation (instead of the finger position).

Ball Motion on Pitch: When set, the ball horizontal position is used to control the pitch of the note (instead of the finger position).

Knobs



SpringSound uses circular knobs (enhanced version of DomesticCat's DCControl github.com/domesticcatsoftware/DCControls). They can be manipulated by a circular gesture or a vertical gesture (choice to set in the configuration menu):

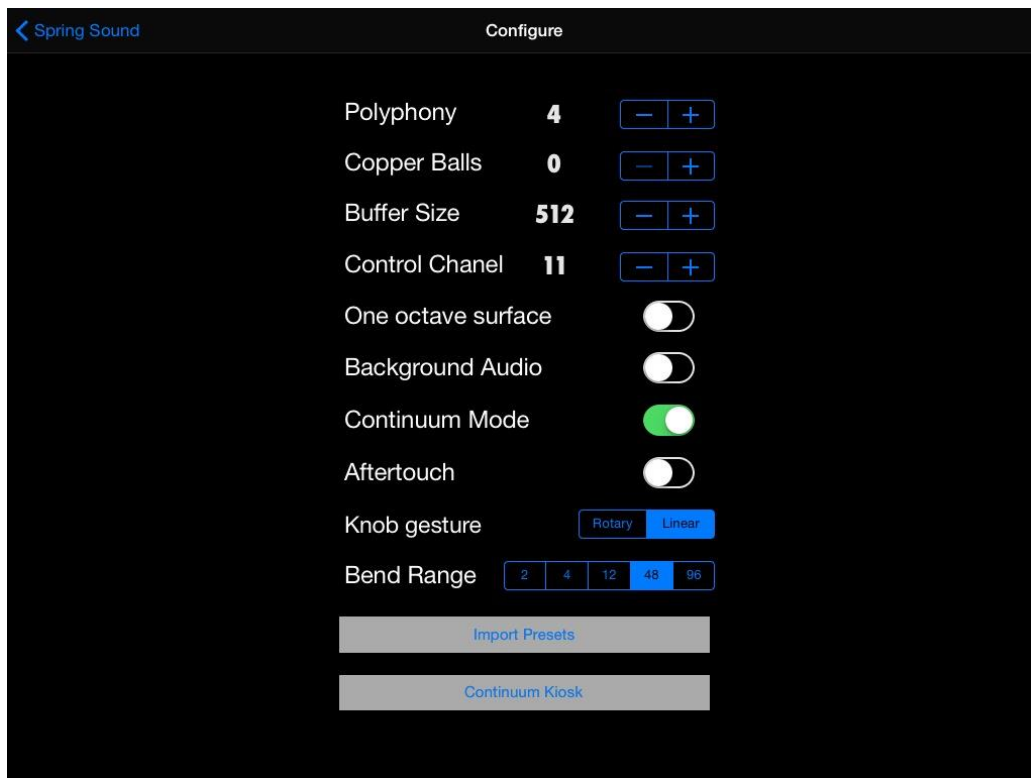
- when set to circular gesture, touch the knob and turn your finger around it to set the value,
- when set to vertical gesture, touch the knob and move your finger up or down it to set the value, are move t left and right for fine tuning

Two additional gestures are available for knobs:

- a double tap sets the value to an intermediate between the minimum and the maximum value
- a triple tap sets the value to the minimum value

Configuration

The configuration screen allows you to set the generic parameters of the app.



Polyphony : This sets the maximum number of note that can be played at the same time. If Polyphony is too high for your device, this may result in a distorted audio or no audio. Here is a set of recommended values:

- iPhone 4 : 3 and Buffer Size of 1024
- iPhone 4S : 5 to 6 and Buffer Size of 512 or 1024
- iPad 2 : 5 to 6 and Buffer Size of 512 or 1024
- iPhone 5 and higher: whatever you want (except if you have 12 fingers on your hand ...).
- iPad 3rd and 4th generation and higher : whatever you want

Note that additional background application or network (mainly GSM) activity may result in distorted audio. Of course, using multiple audio application with Audiobus may also require to reduce the polyphony.

Buffer Size : This sets the internal audio buffer size. It can be set to 256, 512, 1024 or 2048. The lower the buffer size, the lower the latency. Use higher values if you experience distorted audio.

CC Chanel : This sets the MIDI channel that will be used for the remote MIDI control of knobs. The default channel is 11.

Copper Balls : The copper balls are explained at the beginning of the document. You can have 0, 1 or 2 active balls. The Polyphony has to be greater or equal to the active ball number (therefore a polyphony set to 1 requires 0 or 1 active ball, not 2).

One octave surface (iPad only) : By default the iPad playing surface is two octave wide (only one on iPhone).

Background Audio : Select this if you want *SpringSound* to still be active when in background (useful if you want to switch from *SpringSound* to a MIDI sequencer app like LittleMIDI). Remember that if Background Audio is on, even if in background, the app will use CPU resources.

Note that Background Audio is automatically activated when *SpringSound* is used with Audiobus.

Continuum Mode (iPad only) : When unset, this will prevent *SpringSound* from managing special Haken Continuum MIDI message and from proposing the Continuum IOS remote (on iPad only)..

Knob gesture : Set the knob manipulation gesture to rotary (turn your finger around the knob) or vertical (move your finger up or down after touching the knob).

Bend Range : Set the (MIDI) pitch bend range. A set of usual fixed values are proposed. Values 48 and 96 are mainly dedicated to the Continuum. Note that playing on the device screen is also designed to be used with the 48 half tone range.

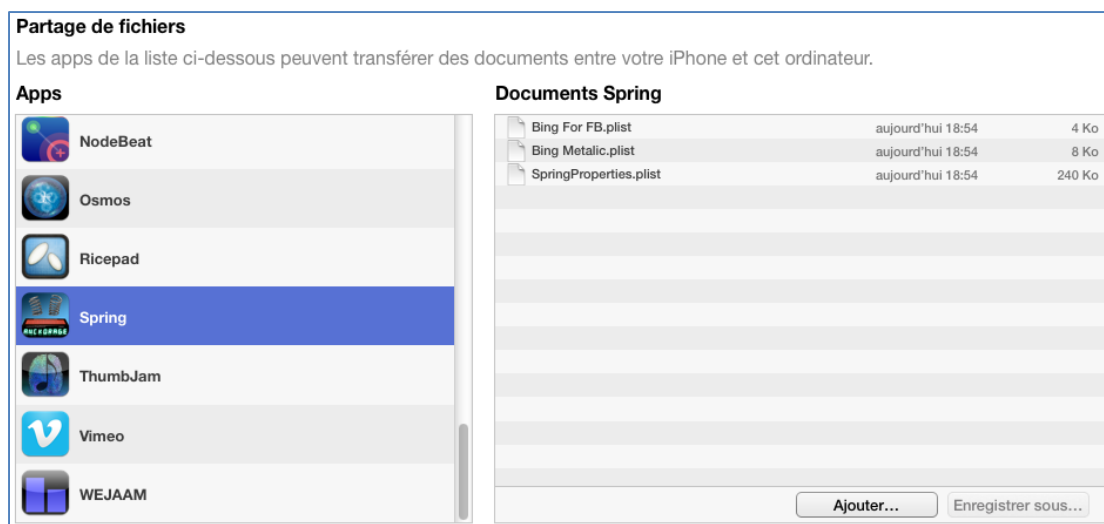
Aftertouch : Aftertouch (MIDI channel pressure) can be used to control the pressure spring (as the Continuum Z axis would do). Set this parameter to On to do so.

Import Preset : *SpringSound* allows you to import and export your presets (sound parameter memory). This functional feature is based on the Apple iTunes File Exchange. If you select Import Preset you will go to the following screen displaying all the presets available for import.



You just have to touch the name of the preset you want to import (you will get a confirmation message). To remove one, select Edit on the upper bar (it will turn to edit) and then, as usual on IOS use the Minus and Delete buttons (select Done to end deletion mode).

To add an additional preset to this list, you need to use the iTunes File Sharing feature: Start iTunes; Connect your iOS device to your computer; select your iOS device from the Devices section of iTunes; Click the Apps tab and scroll down to the bottom of the page; Under the File Sharing section, you'll find a list of apps currently installed on your iOS device that support File Sharing and *SpringSound* will be one of them. Select it to view the list of the files, as on the screenshot below.

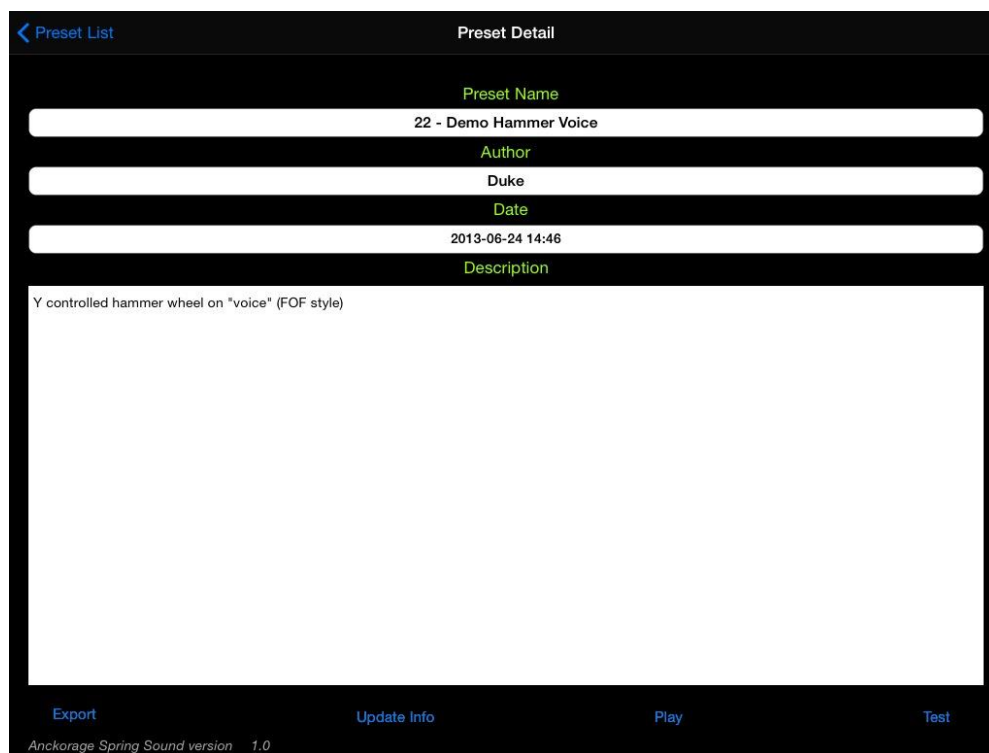


SpringProperties.plist is your all sound bank (you can drag and drop it to your computer to save it, but you won't see it in the import file list of the app. The other files are individual presets generated when selecting Export from the preset detailed view (see below). You can drag and drop these files and mail them to your friends. If some of your friends send you *SpringSound* presets, just drag and drop them in this iTunes window and you will see them in the Import Preset file list of the app.

Preset Management

SpringSound provides a detailed preset management. The presets can be loaded using the rotary picker on the parameter view on iPad, or through the Preset screen which is scrollable and displays all the preset names. When on the Preset screen you still can control *SpringSound* with a Continuum or a MIDI device.





Recalling a Preset is done simply by touching its name: you then go to the Preset Detail view. At this stage the preset is already loaded, and you can play it from a MIDI device. You can:

- test it pressing the Test button (it will play random notes),
- export it to iTunes File sharing (see above)
- directly go to the playing screen by pressing Play
- or change the Description and select "Update Info" to save them

Saving a new Preset is done by tapping **+** at the top of the screen, then enter the Preset name and information_ in the Detail view, and then tap **Save** (or back to Preset from the menubar to cancel ...).

To update a Preset, tap Edit, and then tap on the name of the Preset to update, and confirm (**Ok**) or **Cancel**. Tap **Done** to exit the edit mode.

To delete a Preset, tap Edit, and then tap on the minus sign before the name of the preset and then tap the **Delete** button that appear on the right of the name. Tap **Done** to exit the edit mode.

Note that if a Preset is named **INIT** (in uppercased), it will be automatically loaded at the application startup.

Continuum Remote

The Continuum Remote page provides a full and accurate control of a Haken Continuum fingerboard. Of course you will need to be connected (through a MIDI connection) to a Haken Continuum to be able to take advantage of this page: if not, a warning message will be displayed, and you will be brought back to main menu.

Warning: As the Continuum is having a very heavy use of MIDI messages, you will need to have a good quality MIDI interface between the iPad and then Continuum: some cheap USB-MIDI cables appear to lose some message resulting in text character loss and other and other harms. As example good interfaces are iConnectMIDI2 and 4 (<http://iconnectivity.com>) or iRigMIDI and iRigMIDI2, the first being preferred since it can power supply your iPad (<http://www.ikmultimedia.com>).



Nearly all the Continuum parameters can be set through *SpringSound*'s IOS Continuum Remote:

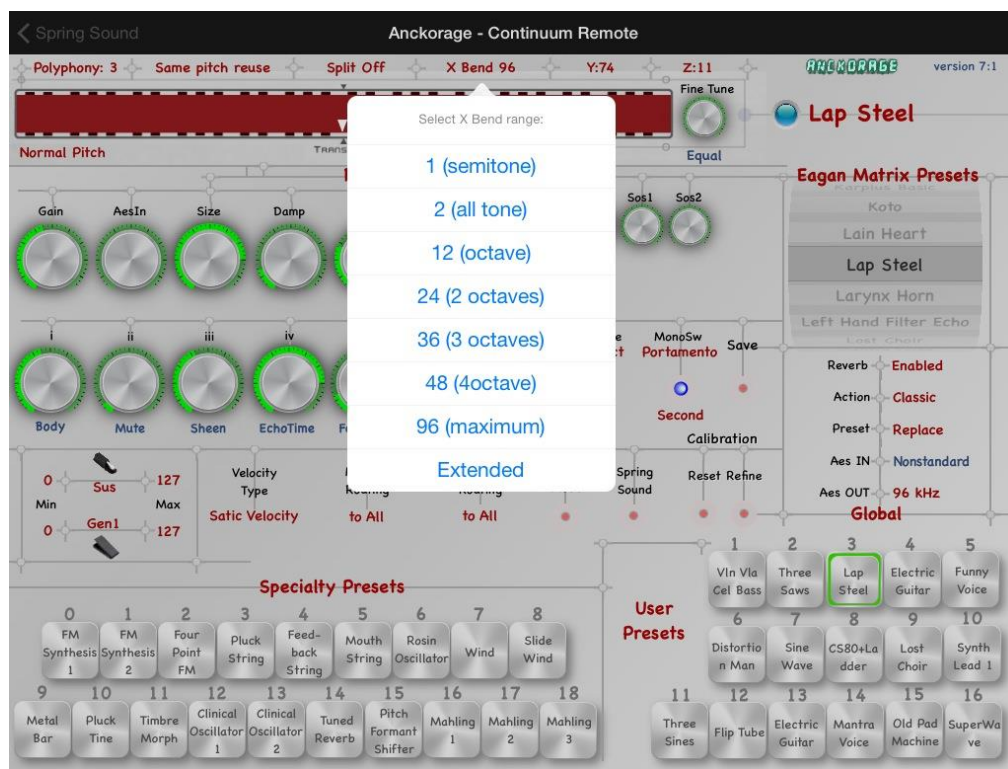
- Specialty preset selection
- User preset selection
- Egan Matrix preset selection
- All sound tweakable parameters (i to iv barrels, Gen 1 and 2 parameters, reverb parameters, main level, etc.)
- All playing parameters (continuous and switch pedal behaviour, tuning and rounding, portamento, etc.)
- Continuum and surface parameters (split, polyphony, MIDI channels and controls, X/Y/W controls, etc.)
- Hardware settings (Calibration, AES parameters, etc.).



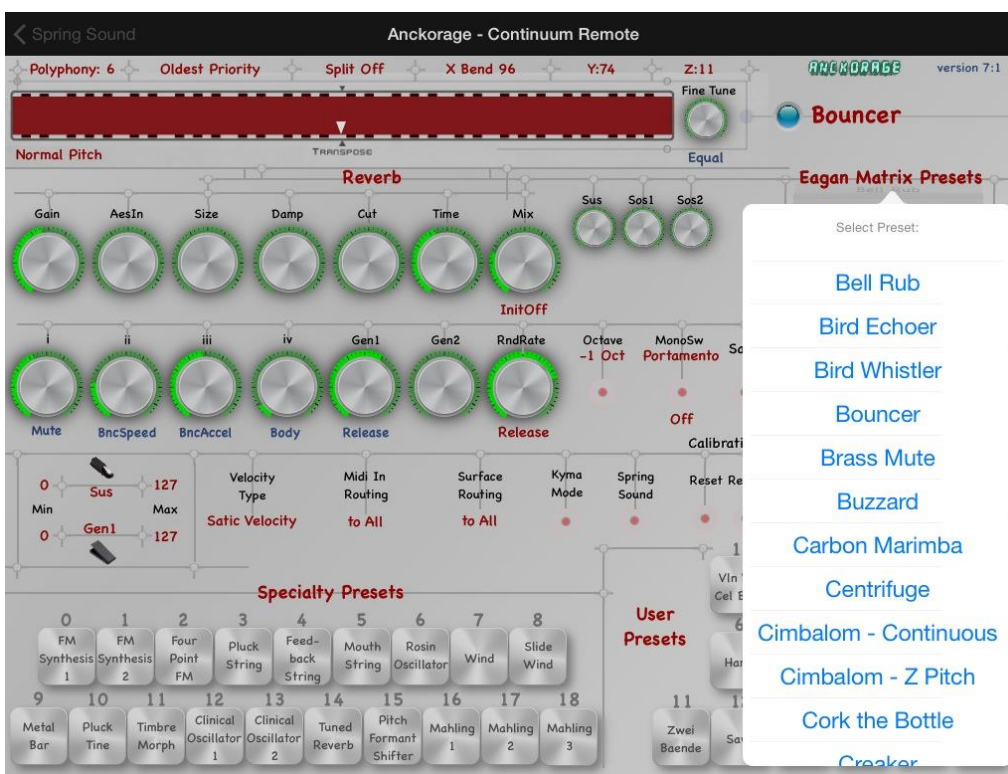
The full detailed and explained list of the Continuum's parameter is available in the Continuum's documentation (<http://www.hakenaudio.com/Continuum/Downloads/ContinuumUserGuide.pdf>).

Note that the Egan Matrix editor is not included in IOS Continuum Remote: sound design for the Hake Continuum needs to be done from the dedicated software editor (Mac and PC).

The following picture illustrates a parameter setting (Bend range).



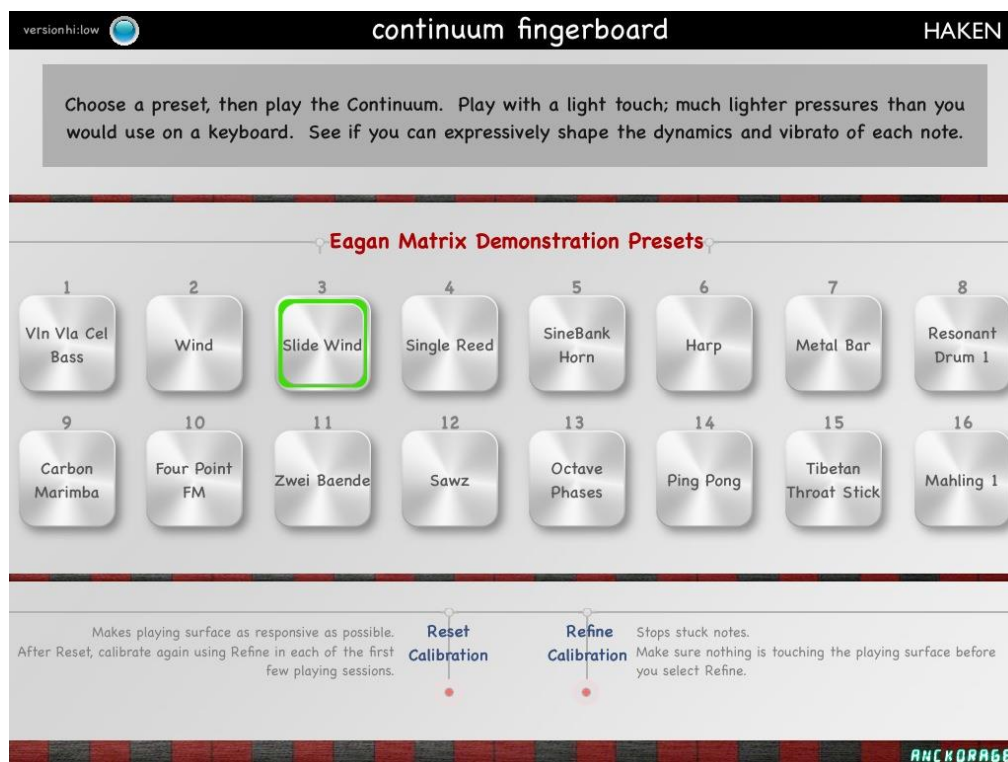
As shown on the picture below, Eagan Matrix preset selection can be done either from a rolling picker, either from a big scrolling list (by tapping on **Eagan Matrix Presets**).



Continuum Kiosk

SpringSound also provide a special Continuum Kiosk mode dedicated to easily demoing the Continuum. This is a very special mode usually not used by Continuum player, but very useful for demo. It is accessible from the Configure Page.

Warning: the Kiosk mode erases all the 16 Continuum's user preset, so make sure to save them before using the Kiosk, in order to avoid any preset loss.



Continuum configuration

In **SpringSound** everything is prepared to work with a Continuum. The only thing to do on the continuum is :

- if you use the MIDI channel pressure for Z, also allow Aftertouch in the Configuration menu
- Configure **SpringSound** and your Continuum with the same 48 or 96 half steps for the pitch bend range;
- Set the note on MIDI velocity to Z (not the fixed 127 value);
- Set the Continuum polyphony accordingly to the **SpringSound** configuration (same values).

The Continuum extended X/Z/Y precision feature is fully supported.

Anckorage (*Anckorage*)

Anckorage is a french electronic music group founded in 1988 by Christophe Duquesne (@Methyam), Nanisound (@nanisound) and Eric Pachot. It's music is freely available from <http://www.jamendo.com/fr/artist/343116/anckorage> Christophe Duquesne is also sound designers and developer, and has used these skills to develop *SpringSound*.



Haken Continuum (www.hakenaudio.com)

Haken Audio
continuum fingerboard

Designed by Dr. Lippold Haken and manufactured by Haken Audio, the Continuum Fingerboard is a musical instrument that allows unprecedented real-time performance control. With a greater pitch range than a traditional 88 note MIDI keyboard, the Continuum offers real-time continuous control in three dimensions for every finger that is placed on the playing surface.



SpringSound is specifically programmed to take advantage of the possibilities of the Continuum.

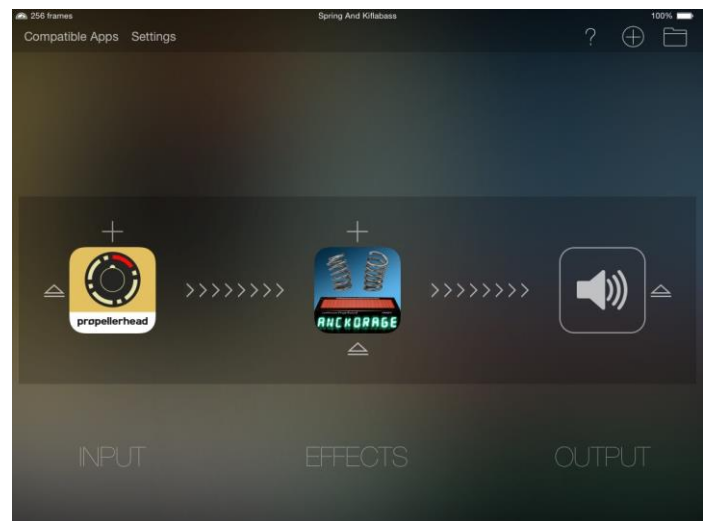
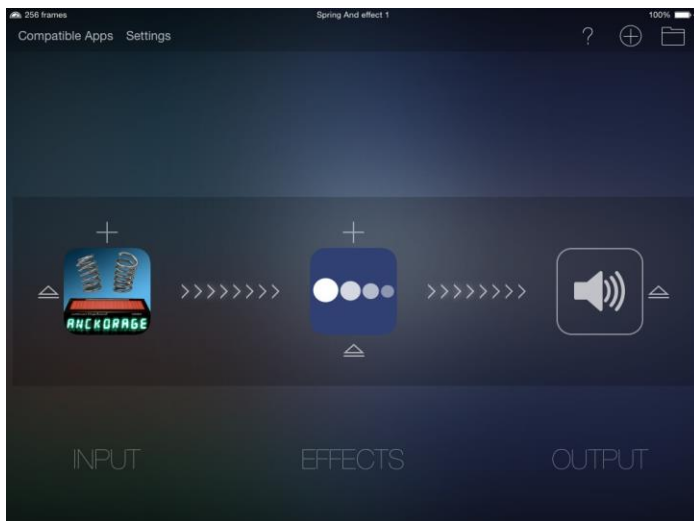
I also would like to take advantage of this documentation for a special thanks to Lippold Haken and Edmund Eagan for their support.

Audiobus (<http://audiob.us>)

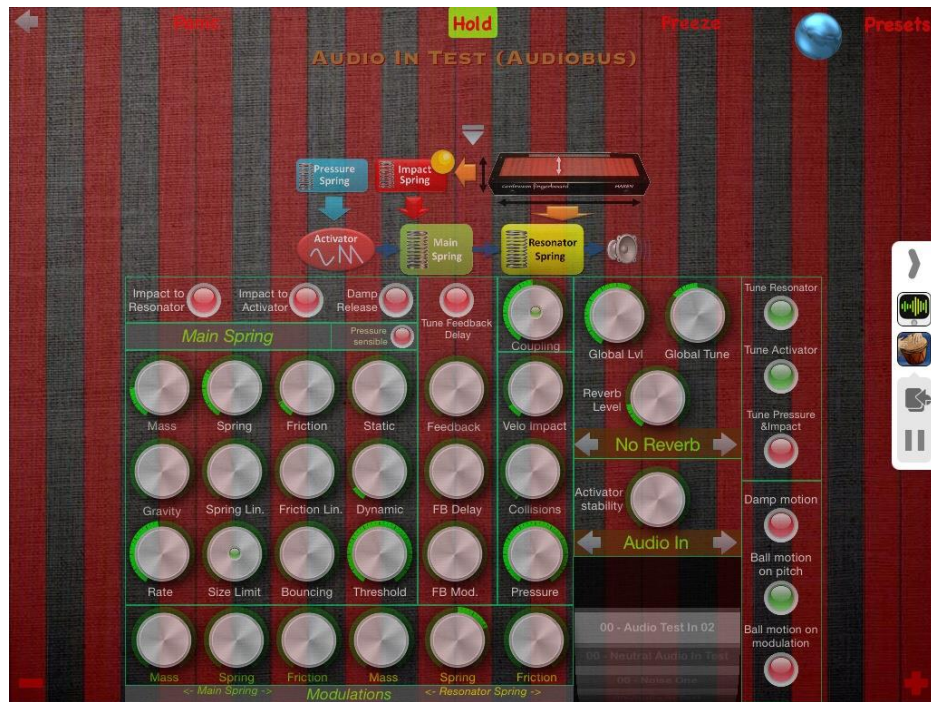
Audiobus allows users to easily connect the output of one Audiobus 2-compatible iOS app into the input of another, playing a synthesizer live into a looper or multi-track recorder, or using one app to manipulate the live output of another.

SpringSound is a fully Audiobus-compatible iOS app, including the state saving mechanism all you to save and recall your configurations.

SpringSound can be used as Audiobus input, effect or output, allowing to use complementary audio effect (like Echopad as shown in the following picture, AltiSpace or Amplitude or any other IOS effect) or to transform the sound of some are the IOS synth you are using (example with Figure below).



The following picture illustrate **SpringSound** connected to DrumJam: you can get easily great result when using **SpringSound** to process drum sounds.



SpringSound is also based on The Amazing Audio Engine developed by Michael Tysen, creator of Audiobus (theamazingaudioengine.com) which is the most advance IOS Audio framework.

Questions and answers, Tips & Tricks

*Why does **SpringSound** has polyphony limitation?*

As **SpringSound** is doing a real world true simulation, it is performing some quite heavy math and physics calculations resulting in high CPU consumption and thus limiting polyphony. But since it is highly optimised, you won't have any trouble with iPad2 and iPhone 4S or higher devices. When used with Audiobus, also take into account the other connected application's CPU requirements, and adjust configuration accordingly. Also remember that other background applications or communications (WIFI and GSM for mails, twitter, etc.) may influence the CPU usage.

Why is the sound out of tune ?

When you change the Mass or the Spring force, it has usually a direct consequence on the tuning. Therefore you may need use the Tune parameter (lower right parameter on spring parameter group) on Main Spring (and Resonator and Impact if they are tunes) to get a tuned pitch.

How two start a new sound design?

There is no single way of starting a new sound design, but there are a few things to avoid:

- Don't start putting all the parameters at a 0 value (low mass, no spring force and no friction usually result in no sound, or even worse: a "broken" physical model producing highly unpleasant sound.
- Avoid starting with feedback activated (Larsen effect is likely to be encountered when tweaking parameters with high feedback)
- Avoid tweaking too many parameters at the same time and concentrate step by step on subsets (usually starting with Mass/Spring force/Friction)

What is this distorted crazy sound I have with high notes ?

As everything is physical in **SpringSound** it is possible to break the model. As it is a software app it won't physically break, but may result in unpleasant sound. As in real life, higher frequency is more likely to break (as higher notes strings for a guitar). In some special cases, if the model is "broken" you may need to use the Panic button to get everything back to normal behaviour.

One small trick often working is to raise up Rate and Mass of the Resonator spring.

I have no more sound (but I had at beginning)!

Physical parameters may result in no sound: for example a high Gravity with a low Spring force, will result in sticking the weighted body connected to the spring to its lowest position, resulting in no sound. One other common reason for getting a sound failure is to have a high Static friction (and even more Static friction + high Mass): if the input force is not sufficiently high, the weighted body will never start moving.

The last thing to check is your polyphony parameter: if too high for your device, this can result in a sound stop.

*Some **SpringSound** sounds are very nice, but hard to play !*

As a real world instrument **SpringSound** is not always an easy to ply play instrument: just try a saxophone, if it is you your first time, you will need some time before being able to get a reasonably nice sound.... **SpringSound** is usually not too complicated but you may need some practice to manage to get proper control of a sound.

My preset doesn't stop playing !

There are two main possibilities for a never ending sound: the Fiction of one spring is too low, or the Feedback is too high! Remember that you have the Panic button to stop everything.