

Density M4L

Ableton Live Instrument and Fx  
granular synthesis devices

user manual v.3.0

[www.apesoft.it](http://www.apesoft.it)

apeSoft © 2019

Starting

- [overview](#)
- [system requirement](#)
- [installation](#)
- [copyright](#)
- [purchase a license](#)
- [update to v.2](#)
- [background](#)
- [GUI layout](#)
- [Quick access](#)

Granular streams

- [parameters](#)
- [settings](#)
- [WavePad](#)
- [WavePad management](#)
- [WavePad display](#)
- [WavePad Snap](#)
- [WavePad mouse](#)
- [WavePad scrub](#)
- [LIVE granulation](#)

Windowing

- [general](#)
- [shape select](#)
- [draw](#)
- [additive](#)
- [audio slot](#)

Snapshots

- [store/recall](#)
- [micro pad](#)
- [transitions](#)
- [snapshots manage](#)
- [subscribe/un-subscribe transition clients](#)

- [transition curve](#)

#### [Snapshots sequencer improviser unit](#)

- [beats cycle](#)
- [time intervals](#)
- [step sequencer](#)
- [output](#)
- [Note and Tick Value](#)

#### [Hyper Vectorial Pad](#)

- [general explanation](#)
- [pad mouse behavior](#)
- [X/Y time](#)
- [miscellaneous](#)

#### [MIDI/OSC I/O](#)

- [header](#)
- [sync](#)
- [widget mapping](#)

#### [Overview](#)

- [save/load device](#)
- [status bar](#)
- [info window](#)
- [float window](#)
- [history](#)
- [acknowledgments](#)

## Starting

### overview

The two Granular Synthesis devices for [Ableton Live](#) (Instrument and Fx), implements respectively a sound file granulator (wav,aiff or mp3) and live-buffer granulator. The first must be loaded this on Ableton Live MIDI track, the second can be loaded either on audio or MIDI tracks like a normal [Ableton Live](#) device Fx.

### system requirement

#### Macintosh

Density requires a Mac PPC or Intel machine running OS X 10.4 or later, and 1 GB RAM.

#### Windows

Density requires a Windows XP/Vista/7 machine and 1 GB RAM.

Density requires Live 9.6 and Max For Live. Details about Max For Live can be found at [Ableton.com](#). This bundle will only work in [Ableton Live](#) (not in the MaxMSP application).

Max for Live puts the power and potential of Max/MSP inside Live. Create all the instruments, effects and extensions you've ever wanted. Go beyond the common and predictable, and transcend the limits that conventional tools impose. Build completely unique synths and effects, create algorithmic composition tools, or fuse Live and controller hardware into radical, new music machines. Join a society of makers and share ingenuity.

Max for Live was co-developed by [Ableton Live](#) and [Cycling '74](#).

Density uses QuickTime in order to read correctly media files (including MP3), therefore QuickTime must be installed on your system. On Windows we recommend a complete installation of QuickTime choosing all optional components.

### copyright

This program is copyright shareware and it is not freeware.

You can download the unregistered version of the program and give it to your friends or to any other person as long as for no charge. This program cannot be distributed in shareware compilations CDs without prior written approval from the author.

No responsibility is taken for any damage or losses caused by this package.

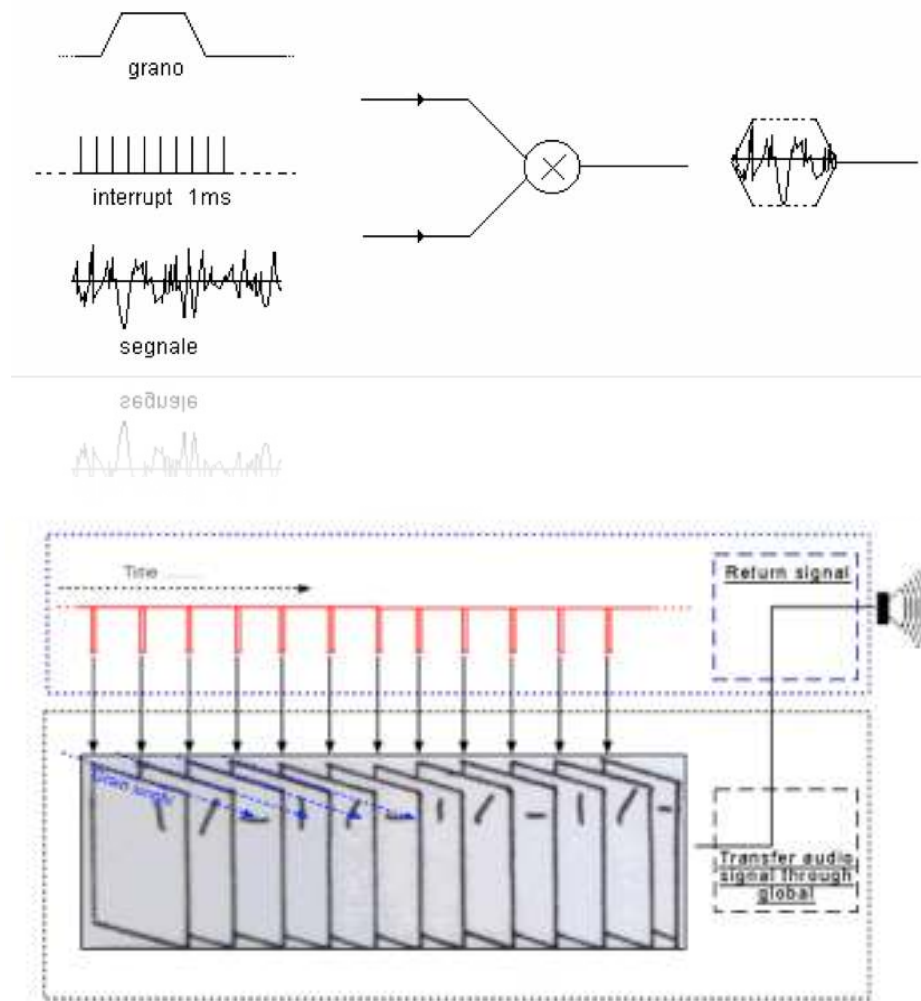
All program trademarks belongs to its respective author.

## background

The first official release of Density (2001), developed in [Csound](#) language and based on [Eugenio Giordani](#) 's GSC4 (Granular Synthesis for Csound). GSC4 was the first patch for granular synthesis on Csound implementing [Barry Truax](#) model.

Density.m4l can generate thousands of grains dynamically, I preferred this way, rather than a fixed number of "voices" (oscillators). The overlapping factor grains, depends only on the actual CPU power. Thus you have not limits in grains number for second (density).

Below an easy granulation model.



## GUI layout

The most important Density.m4l parameters are placed inside of only one window. Density.m4l GUI (Graphic User Interface) is divided in two parts, WavePad, Granular parameters, Snapshots-presets (from left to right). and WINDOWING MODULE (most right). Many controls are accessible via pop-up menù. WINDOWING generate and edit waves prototypes shapes, employed for envelope granulation.

## Density Instrument



## Density Fx



## Quick Access

You can quickly open (From top to bottom)

- Snapshots sequencer improviser unit;
- Hyper Vectorial Pads;
- OSC I/O mapping;
- About;



## Granular parameters / Snapshot Preset / Windowing Module



## Granular Streams

### parameters

In this section we will examine the Granular Synthesis parameters. Please refer to web for more explanations about granular synthesis.

You can interact with Density parameters on the GUI (Graphic User Interface) in the follows ways:

- 1) Click and drag (scroll) on vertical sliders (or dials) to change the value, (see [mousing](#) mode for more details);
- 2) Holding down the Command key (Macintosh) or the Control key (Windows) while mouse scroll, for precise value control;
- 3) Select a widget clicking on the name and use keyboard up/down arrows;
- 4) Drag up/down the display value (the number box below each parameter);
- 5) Click on the number box value, enter a numeric value and press enter key;
- 6) Send a MIDI CC or Ableton Live automation, see [Ableton](#) user manual;

The granulation engine is updated every new generated grain, the frequency of grains scattering depending from density parameter, expressed in Hz (grains for seconds).

The vertical sliders provides a deterministic values, while the dials (knobs) provides random values. Summing the two values (deterministic-random) will be produced a value that is passed to the granulator.

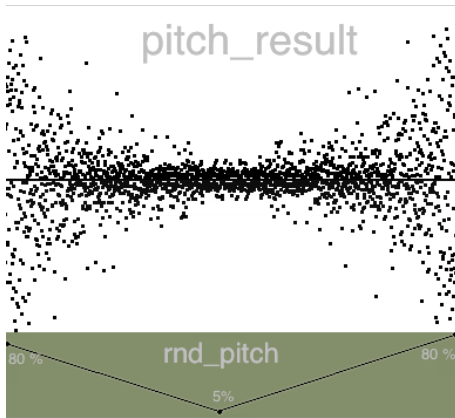
In the following example, the pitch value (deterministic) is rescaled with rnd\_pitch (random):

$$\text{pitch\_result} = \text{pitch} + ( \text{birnd}(\text{rnd\_pitch} * \text{pitch}/2) )$$

Where birand is bipolar random generator. The random range is expressed in %, therefore 0 % it means only deterministic value, 100 % pitch\_result can vary in the range:

$$\text{pitch\_result} = \text{pitch} \pm \text{pitch}/2$$

In the example rnd\_pitch (green frame) starting from 80%, is reduced to 5 % in the middle.



With a great `rnd_pitch`, each grain can have an high random pitch mask range. Carrying `rnd_pitch` toward lowest values, the random pitch mask is reduced until 0 % which means no random pitch grain scattering.

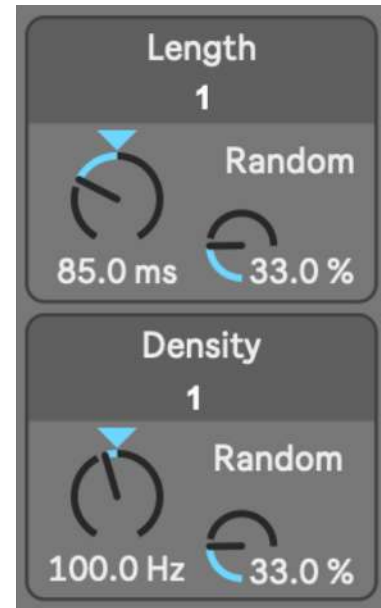
## CONTROLS

**Length** = grain duration in milliseconds;

**Random Length** = random range deviation, in % of length;

**Density** = grain for seconds in Hz;

**Random Density** = random range deviation, in % of density;



**Pitch** = detune factor: value 1: original pitch, value 0.5: octave down, 2 octave up;

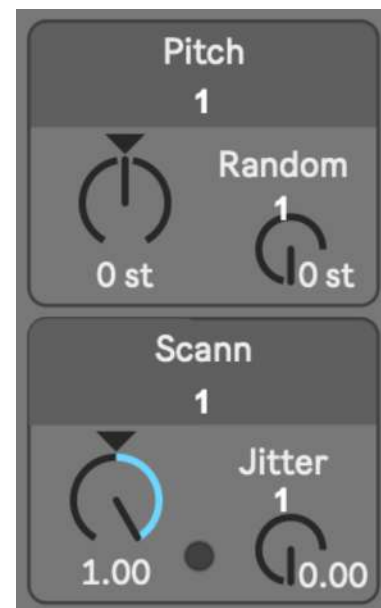
**Random Pitch** = random range deviation, in semitones;

**Scann** = playback scanning ratio:

- value 1: original time,
- value 0.5: half-tempo (time stretching),
- 2: twice-tempo (time compressing);
- 0: freezes playback at the current location;
- negative values rate: backwards play;

**Freeze button** = (small button in the middle of scanning control) switch between the value 0 (freezes) and the last control value;

**Jitter** = randomly moves the time-pointer around the current value (scanning or scrubbing), the value is expressed in milliseconds;



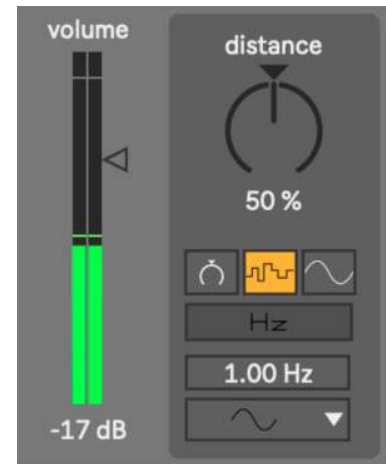
**Volume** = grains amplitude in decibel;

**Pan** = left/right grains distributions on the stereo front (only pan = manually);

**Distance** = stereo field width;

**Panning Mode** = select between:

- pan (manually)
- jittering (random)
- lfo (low frequency modulation);



**Beat/Hz** (only lfo) = toggle between beat/Hz; sets the panning duration (left right azimuth) expressed in note-ratios or frequency (Hz). Beat mode is synced with the Ableton Live global transport time, you can change global time (BPM<sup>1</sup>) from Ableton Live transport.

**Beat Menu** (only lfo) = each item switch represents the Relative-Tempo expressed in note-ratios; you can change BPM (Beat per minute) from the Ableton Live transport.

N.B. you need activate the Ableton Live transport master clock.

---

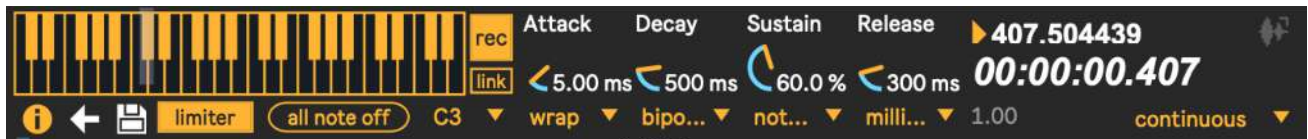
<sup>1</sup> **Beats per minute (BPM)** is a unit typically used as a measure of tempo in music.

The BPM tempo of a piece of music is conventionally shown in its score as a [metronome](#) mark, as illustrated to the right. This indicates that there should be 120 [crotchet](#) beats ([quarter notes](#)) per minute. In simple [time signatures](#) it is conventional to show the tempo in terms of the note duration on the bottom. So a 4/4 would show a [crotchet](#) (or quarter note), as above, while a 2/2 would show a [minim](#) (or [half note](#)). In compound time signatures the beat consists of three note durations (so there are 3 [quavers](#) ([eighth notes](#)) per beat in a 6/8 time signature), so a dotted form of the next note duration up is used. The most common compound signatures: 6/8, 9/8, and 12/8, therefore use a dotted crotchet (dotted quarter note) to indicate their BPM. (Wikipedia)



## settings

Streams setup:



- Through the keyboard you can active/de-active granular streams polyphonically, you can also set the ADSR (attack decay sustain release) event envelope. You can either use the MIDI message note on/off to trigger streams remotely.
- From left to right, number box (6 in the image), sets general Synth polyphony;
- i toggle, enable/disable linear interpolation for sample and envelope oscillators;
- left arrow, set the granulator to read samples backward, you cannot confuse it with a negative scanning ratio value. This setup parameter, set directions reading samples inside of the grain, therefore you can have a normal time scanning with an inverse grain time;
- The number box (64 in the image) change the grains overlapping factor (maximum numbers of simultaneous voices). Default is 64 the max value allowed is 512;
- limiter enable/disable the peak-limiter which allows for the specified control of signal amplitude.
- all note off turn off all playing notes by sending a message to each instance with a playing note. The message consists of the MIDI pitch most recently received via the keyboard note or midinote message followed by a 0 (meaning zero velocity or note-off);
- C3 menù is the keyboard octave transposition, C3 key is gray emphasized;

## WavePad

WavePad is the audio buffer used to load the audio file. You can load an audio file from pop-up menù (load/replace item), or dragging a sound file directly on the pad. Supported audio files: WAV, Next/Sun, AIFF, MP3<sup>1</sup> file informations will be displayed on status bar below the WavePad.



On the right side of the WavePad, you can choose the type of action: select, loop, move and draw. The tools select (from the top) and loop allows you to scroll the buffer manually (scrubbing) and settings grain length.

If automatic scanning parameter is at work (unfrozen ratio != 0), clicking and scrubbing on the WavePad will disable temporally automatic-scanning as soon as you press mouse down button. The automatic playback scanning it will be resumed at the current position when you release mouse up button.

**loop** (second from the top) = unlike to select, controlling also the grain length. Drag it horizontally for time scrubbing and vertically for grain length.

**move** (hand) = allows you to select a part (zoom) of loaded audio file. N.B. the selected part it will be employed for current granulation. Automatic scanning will read cyclically (loop wraparound) the current selection;

**draw** (pencil) = allows you to draw directly on the pad.

<sup>1</sup> Density uses QuickTime to convert a media file (including MP3 files) into the sample memory of a wavePad and requires that QuickTime be installed on your system. If you are using Density on Windows, we recommend that you install QuickTime and choose a complete install of all optional components.

## WavePad management

**load/replace** = open a file browser to load an audio file on the selected slot (aiff, wave mp3 supported). Also you can drop a valid file on the wave pad, if it's unsupported, a message will be shown;

**savetofile** = export the contents of the buffer as Wav or Aif audio file;

**openbuffer** = opens the WavePad buffer window, or brings it to the front if it is already open. The window is resizable but not editable, you can scroll the mouse over the buffer window;

**crop** = will trim the audio data to the current selection. It resizes the buffer to the selection length, copies the selected samples into it, and displays the result at default settings. The buffer is erased, except for the selected range. This is a "destructive edit," and cannot be undone;  
**clear buffer** = erases the contents of buffer;

**lastsel** = it causes the selection start and end points to revert to their immediately previous values. This is helpful when you are making fine editing adjustments with the mouse and accidentally click in the wrong place, or otherwise cause the selection to change unintentionally. Repeated undo commands will toggle between the last two selection states;

**zoom out** = resets the entire Hzoom (horizontal zoom) time display;

Clicking Normalize to scale the sample values in the WavePad buffer (0 dB), so that the highest peak matches the value of 1. Scroll number on the right to manually adjust value.

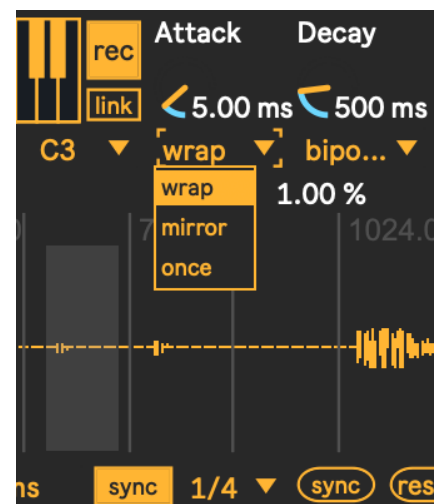
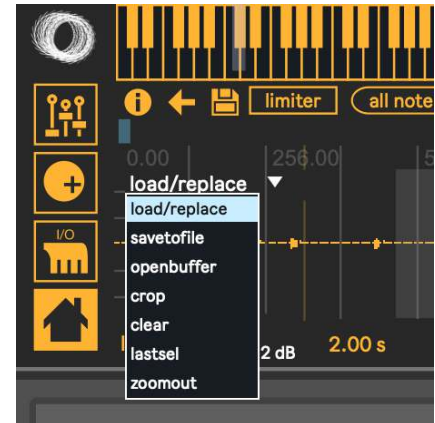
e.g. normalize = 1.50 scale the sample values in the buffer so the highest peak matches the value given by the argument. This can cause either amplification or attenuation of the audio, but in either case, every value is scaled. When a sound file is loaded, default value is 0, it means original wave amplitude, value greater than 0 are saved and loaded in the project. N.B. Normalized value of 0 can be interpreted like:

1. original file amplitude if you have not change it before,
2. normalizing at 0 value (i.e. 0.001 amplitude gain, -60 db).

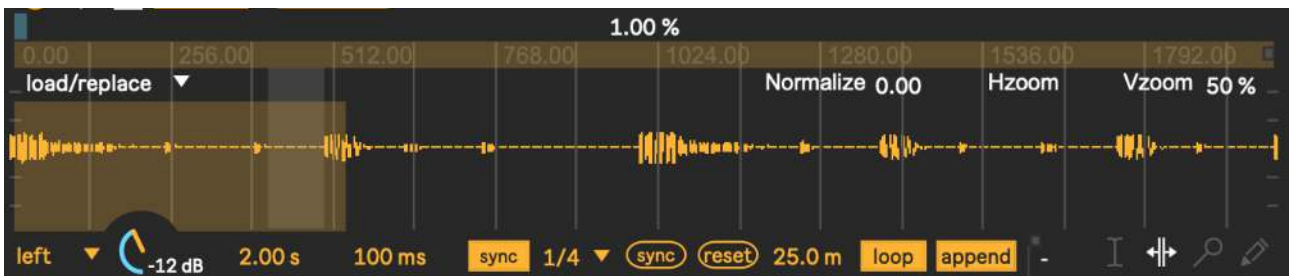
**wrap** = turn around waveform length in auto-scanning mode (scann > 0);

**mirror** = read forward/backward the waveform;

**once** = reads the waveform one once, and turn off all notes when the end of waveform is reached;



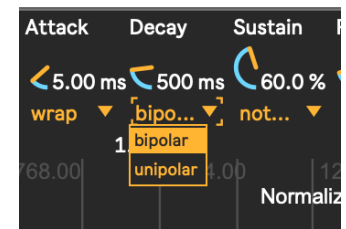
## WavePad display



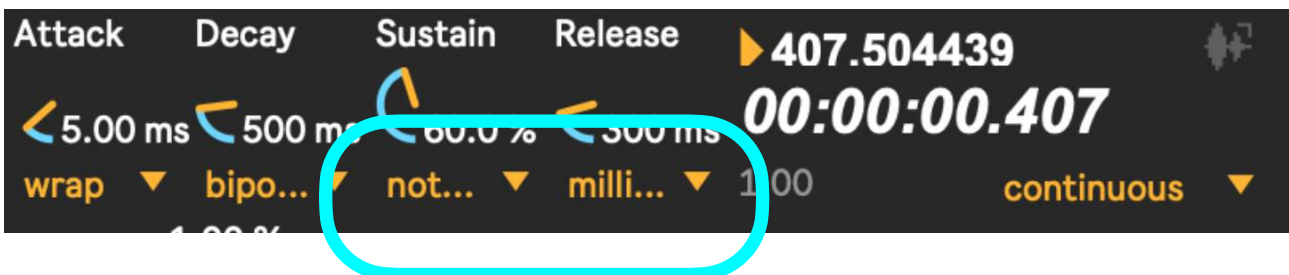
Clicking Hzoom to reset horizontal zoom (entire time display of file);

Clicking on Vzoom you resets vertical amplitude zoom (default 50 %). Mouse scrolling on right number (50 %) to set an amplitude zoom amount. N.B. unlike Normalize, Vzoom is not a destructive amplitude rescaling but only a graphics rescale.

bipolar menu switch to unipolar/bipolar view mode, unipolar mode shows values between 0 ÷ 1 while in bipolar shows values between -1 ÷ 1;



## WavePad Snap



You can set the WavePad snap mode selection range:

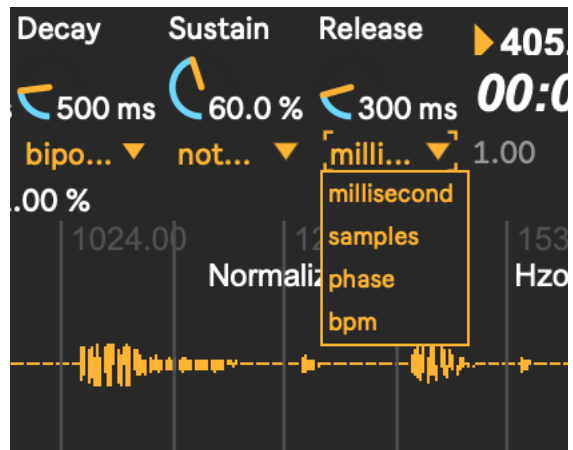
Snap causes the start and end points of the selection to automatically move to specific points in the buffer, defined by the snap mode:

- nothing disables snap to allow free selection. This is the default;
- grid specifies that the selection start and end points (grain length) should snap to the vertical grid lines;
- zerocrossing instead of snapping the selection to a uniform grid, this mode searches for zero-crossings of the [buffer](#) data;

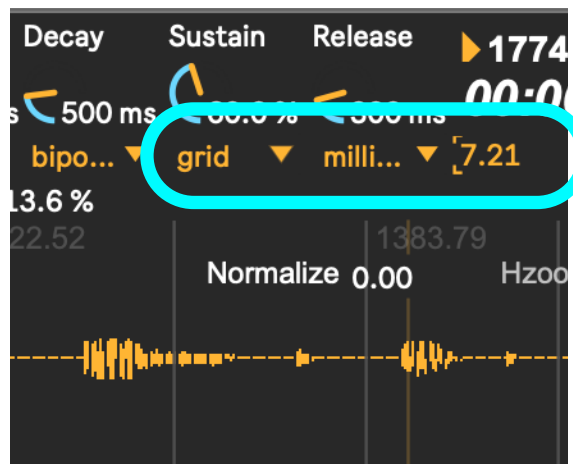


The seconds menù from the top (millisec...), sets the unit of time measurement used by the display:

- **milliseconds** = sets the display unit to milliseconds;
- **samples** = causes time values to be shown as sample positions in the target buffer. The first sample is numbered 0, unless the display has been shifted by the offset message;
- **phase** = causes time to be displayed according to phase within the buffer, normalized so that the 0 refers to the first sample, and 1 refers to the last; N.B. grid values for phase, must be normalized ( $0. \div 1.$ ), (see later);
- **bpm** = specifies beats per minute as the time reference unit, relative to a master tempo and number of beats per bar, both of which you can set with the bpm message.



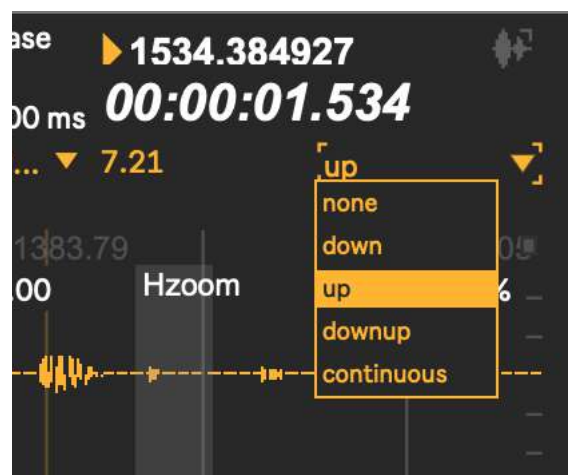
The numbox (3.00) specifies spacing of the grid lines for current unit of time measurement used by the display. N.B. grid values for phase, must be normalized ( $0. \div 1.$ )



## WavePad mouse

The way that the WavePad responds to mouse clicking and mouse dragging varies according to:

- **none** = selection start and end values are not sent in response to mouse activity;
- **down** = causes the current selection start and end values to be sent only when you click inside the WavePad;
- **up** = causes selection start and end to be sent only when you release the mouse button, after clicking inside the WavePad;
- **downup** = causes selection start and end to be sent both when you click inside the WavePad, and when the mouse button is released;
- **continuous** = causes selection start and end to be sent on click, release, and throughout the drag operation, whenever the values change;



## WavePad scrub

Top left we have the fine scrub and below the elapsed time display (hh: mm: ss: ms).



Clicking on waveform icon you will open buffer window (see menu buffer management).

**fine scrub** = clicking and dragging up and down on the number box with the mouse moves the displayed value up and down, and outputs the new values continuously. In the float number box, dragging to the left of the decimal point changes the value in increments of 1. Dragging to the right of the decimal point changes the fractional part of the number in increments of 0.01. The numbers can be entered into a number box by clicking on it with the mouse and typing in a number on the computer keyboard. Typing the Enter keys or clicking outside the number box;

Buffer window scrub (mouse down button), behave like scrub on WavePad (see above for more details).

N.B. WavePad zoom has a time zoom selection, opening the buffer will display the entire contents of the buffer.

## LIVE granulation



Density FX captures a mono input from channel, writes into a buffer. The granulator engine read from the buffer, you can use all granulation parameters in real-time.

LIVE granulation involve a realtime input (mono) capture from ADC (Analog to Digital Converter). Samples are written cyclically (ring buffer) in the WavePad buffer. All the granular parameters work also in real-time for LIVE although LIVE granulation it need of a “samples gap” (i.e. a delay between the writing/reading indexes) in order to trigger LIVE granulation.

The physical input channels ADC is set by the [Ableton](#) mixer (see [Ableton](#) user manual for explanations).





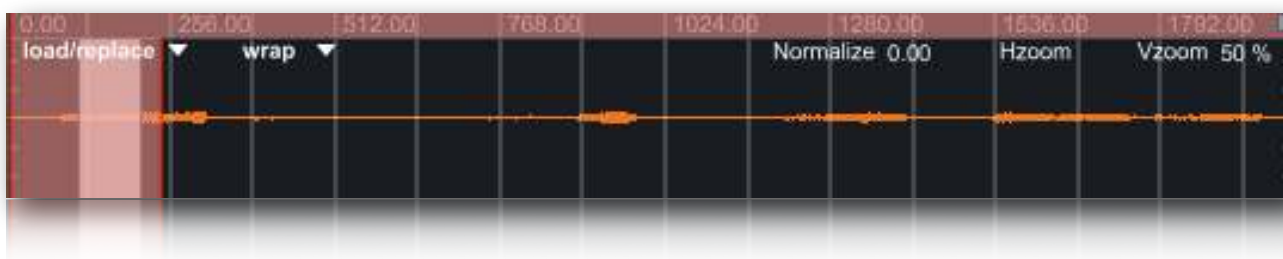
From left to right we have input channel select and gain in dB, you can monitoring the input level and rescale the input amplitude signal, you can also select left or right channel from input menu (ADC Analogic to Digital Converter). Live granulation is mono, you can also mix both channels (l/r-merge).

buffer length (2.00 s), in milliseconds for LIVE buffer. In the example we allocated a buffer that contains 1.450 milliseconds;

Turning on LIVE rec, you will see on the WavePad two indexes: red is writer index, while read is scrub index (gray). If scanning is 1, the two pointers will move at constant speed.

The time distance that separates them is “delay gap”, you can change delay gap from numbox (100 ms default) write/read, settings a new delay time in milliseconds between the write/read indexes (i.e. the granulation secondary buffer);

If link is enabled, when you trigger a keyboard event, the LIVE rec is automatically turned on.



The synchronism between the two indexes (write/read) can be done manually scrolling reading index on the WavePad, clicking sync button or in automatic enabling sync (yellow toggle).

**sync toggle** =correlates write/read indexes. Read index it will be pulled by the write.

The time of synchronization is expressed in BPM, each item of beat menu (only lfo), represents the Relative-Tempo expressed in note-ratios; you can change BPM (Beat per minute) from the Ableton Live global transport.

N.B. you need activate [Ableton](#) Live transport, see [Ableton](#) manual for further details.



reset (from left), restore write in whole WavePad, zooming out view. If loop is disabled, it will trigger one-shot buffer write;

fade in/out (25.0 ms) in milliseconds, you can set a fade in/out to avoid clicks every time you turn on/off the capture LIVE;

If loop write mode is enabled, when write index reaches the wavePad right limit, write index wrap around of the WavePad, if disabled, capture will be stopped when write index reaches the end of WavePad. Loop is enabled by default;

If append mode is enabled, every time you start LIVE capture the write index continues from where it was last stopped. If disabled the capture starts from beginning. Append is enabled by default;

## Windowing

### general

The windowing module generate some classics envelopes and prototypes used for smoothing the grain amplitude. The currently selected window, it will be employed for the grain envelope during granulation.



You can also deform some envelopes shapes like gauss, curve or additive, draw a new shape freehand or through draw prototype. Also you can load an audio file from disk (until 6 audio files, Aiff Wav or Mp3 supported). Almost all Windowing WavePad functions are identical to [Granular Streams](#) WavePad.

See [WavePad](#), [WavePad display](#), [WavePad Snap](#), [WavePad mouse](#) for more details.

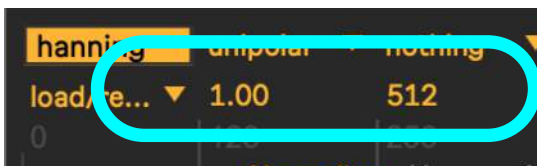
Unlike Granular Streams WavePad, the Windowing WavePad buffer management have two additional functions:

### reset shape

When you select from menu the first of them, current envelope or shape will be restored at the original shape.

When you select reset all shapes, all prototypes and envelopes they will be restored at the original shape.

resetshape and resetallshapes, they work on the sixteen envelopes/prototypes restoring originals shape. While clear work either on the sixteen envelopes/prototypes either on the six sound files slots, writing 0 in the all buffer samples.



Windowing has an important parameter that allow you to chose a size in samples for prototypes or envelope shape. You can also resize buffer dynamically, entering a new value expressed in samples (512 default).

N.B. when you resize a shape, you will reset original default shape.



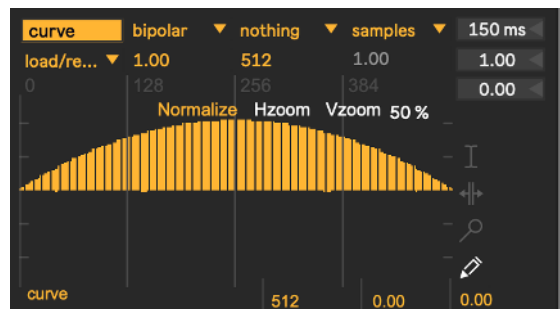
## Shape Select

There are 16 pre-generated shapes:

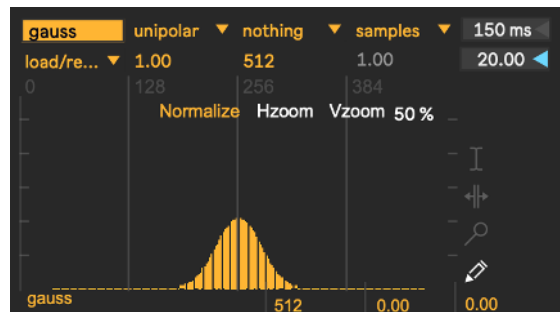
1. rectangle,
2. blackman,
3. curve,
4. gauss,
5. hamming,
6. hanning,
7. draw,
8. sine,
9. cosine,
10. triangle,
11. saw,
12. square,
13. sigmoid,
14.  $\sin(x)/x$ ,
15. random,
16. additive

Some of them have additional controls parameters (bold). Selecting one of them, their additional parameters will be shown.

**curve** shows three box-number, on the top (top right 150 ms) we have deformation update rate in milliseconds. Since these parameters work in real time, you can set a speed limit time to avoid cpu overhead. The other two parameters (1.00 and -0.72) sets middle/border curve deformation.



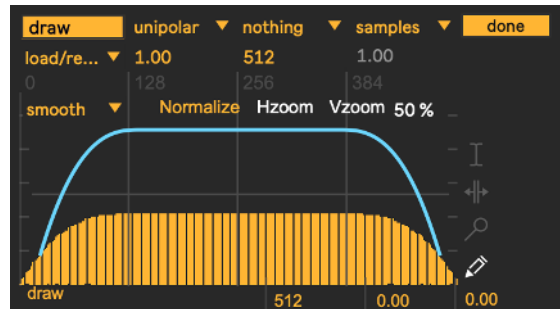
**gauss** - Like curve, gauss show on the top the speed limit update rate and only a "Gaussian standard deviation" parameter.



## draw

In Density, the grain envelope shape default is draw (bold italic), when selected appear yellow button (draw/done).

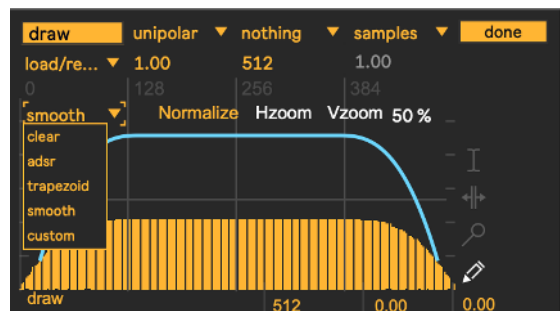
This allows you to superimpose at the WavePad a function break-points where you can track the envelope segments.



You can add new break-points by clicking on the function superimposed pad or shift + click on a specific break-point to remove it.

Default shapes are available, adsr (Attack Decay Sustain Release) and a trapezoid (default grain envelope).

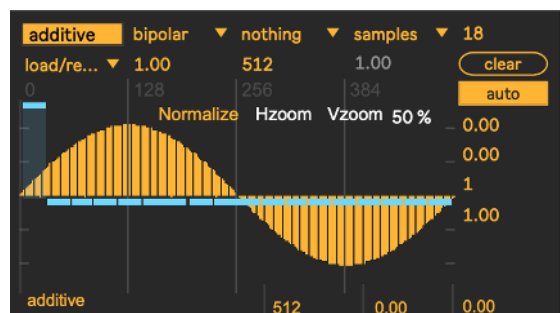
You can switch between unipolar/bipolar automatically when you change view perspective in windowing WavePad, (see [Granular Streams WavePad display](#) for more explanations).



## additive

additive generates a composite waveform through weighted sums of sinusoids. It works similarly to Csound Gen19. Each slider in the additive pad, controls a specific partial amplitude.

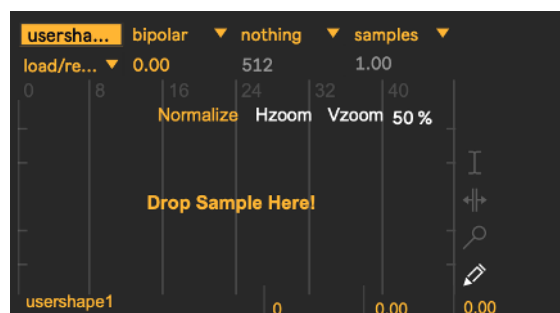
You can change the total harmonics number for the spectrum (top right numbox 18), clear all amplitude contributes of spectrum (bringing to 0).



Toggle on auto, you will enable amplitude auto-rescale, thus the resulting shape it will be always normalized to one. N.B. phase and offset are normalized ( $0 \div 1$ ). The two below num-box of gen button, sets the fine amplitude of a specific partial. You need selects a partial number (1 in the example), enter the amplitude value (0.5 in the example) at last click on gen to apply change.

## audio slot

In the shape select menù, after the sixteen envelopes/prototypes you will see six empty slots named: usershape1, usershape2 etc.. You can employ them to load until six audio files from disk, just like [Granular streams WavePad](#). See for further details. If cannot load, or drag, a sound file in a shape slot, you must employ one of the six available slots.



## Snapshots

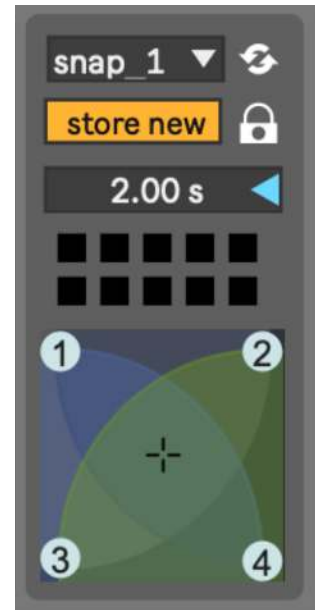
### store/recall

A snapshot is a photo of graphical interface (GUI parameters) in the current state. Each snapshots module can store up to 24 snapshots.

You can make transitions between two or more snapshots, you can subscribe or unsubscribe determinate Density parameters.

When you find an Interesting "sound" you can store new snapshot just shift + click on a button in the snap-pad, thus the preset button will light orange. You can recall a snapshot by simply clicking on the snap-pad buttons; all the widgets subscribed (see later) will be restored at current snapshot value, immediately.

Alternatively a snap can be stored from pop-up menù, a message box will ask you to digit a snap number in (see below for more explanations).



### micro pad



The micro-pad's goal is to obtain intermediate values between four snapshots (interpolation). You can configure four nodes (snapshots).

**refresh** (highest in the image) enable/disable auto-recall snapshots, see below for more details.

**locked** (lowest in the image): nodes: the mouse can only edit the nodes position and size. unlocked: slider; the mouse can only changes the slider location.

### transitions

Transitions are a linear interpolation between two snapshots, with one time duration.

You can start a transition only from the menù recall item.

Is important to understand that the transition occurs from the current parameters positions, toward the selected snapshot.

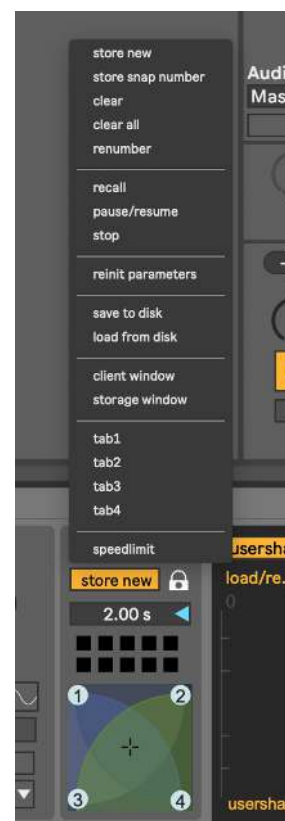
actual GUI widgets positions >>> (toward) selected snapshot (time duration)

You can change the transition-time (2.00 sec in above image). The transitions can occur simultaneously on the streams and/or on the main. If auto-recall (see above) is enabled, selecting from the menu a snapshot slot to start the transition.

•

## snapshots management

- **store new** snapshot using the next empty preset slot;
- **store snap number**, enter a number or a list separated by space, to store;
- **clear** current selected snapshot;
- **clear all** snapshot, a message box will ask you to confirm;
- **renumber** sort all snapshots stored into consecutive, beginning with 1;
- **recall**<sup>1</sup> start a transition toward current snap number;
- **pause/resume** current transition (if occur);
- **stop** transition cancel current transition (if occur);
- **reinit parameters**, resets subscribed parameters at default value, only for the current streams or main<sup>2</sup>;
- **save to disk** current snapshots-bank;
- **load from disk** current snapshots- bank;
- **client window** open clients list to subscribe, unsubscribe, setting interpolation type and more;
- **storage window** open storage window displays any stored presets;
- **tab1, tab2, tab3 and tab4** open transition curve window, see [transition curve](#) below;
- **speedlimit** set update time limit for transitions and micropad nodes interpolations.



N.B. all the presets are saved in the Ableton Live project or device, however, you can manage individually save/load snap-bank, for example useful for exchanging presets between the streams.

## subscribe/un-subscribe transition clients

The two windows client and storage, are both non-interactive:

	Name	Priority	Interp	Data
✓	density	0	linear ⇅	34.324196
✓	distance	0	linear ⇅	0.858665
✓	jitter	0	linear ⇅	1.303557
✓	length	0	linear ⇅	583.201843
✓	pan	0	linear ⇅	0.
✓	pan-freq	0	linear ⇅	1.000049
✓	pitch	0	linear ⇅	2.437552
✓	rnd_dens	0	linear ⇅	23.702206
✓	rnd_len	0	linear ⇅	21.508684
✓	rnd_pitch	0	linear ⇅	0.
✓	scann	0	linear ⇅	0.651778
✓	volume	0	linear ⇅	-6.
✓	scrub	0	linear ⇅	1.

client window (accessible from the client window menu item, see above) shows the current subscriptions widgets list (ever header name), priority, interpolation and data belonging to subscribed widgets.

A few types of interpolation can be changed:

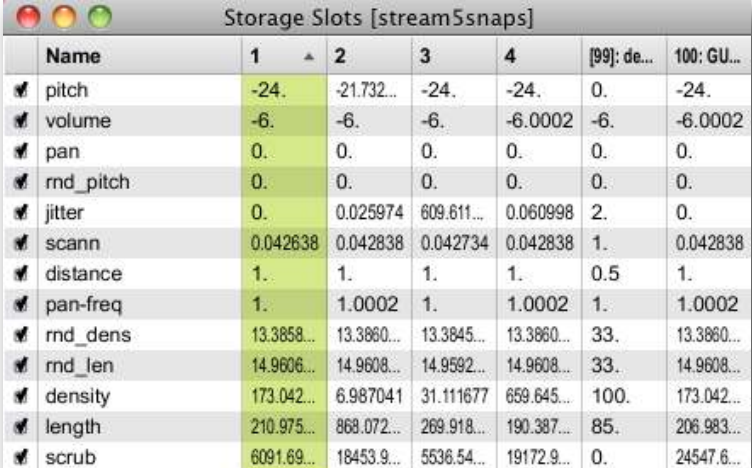
- off: no interpolation;
- linear: Linear interpolation. Presets recalled will be interpolated using a standard linear algorithm.
- threshold: Threshold. Takes optional argument (float), which sets the threshold.

<sup>1</sup> you can enable or disable auto-recall, if enabled selecting a snap from the menu, it will start the transition, if disabled it will be **only selected**.

<sup>2</sup> gsnap will restore all Density.m4l subscribed parameters at default values (granular parameters and setup parameters).

- Presets recalled will recall data from the first preset specified when the fade amount is below the threshold, and will recall data from the second preset specified when the fade amount is greater than or equal to the threshold. e.g. threshold: 'fade' < thresh = value a; fade >= thresh = value b;
- inverted threshold: Inverse threshold. Takes optional argument (float), which sets the threshold. Presets recalled will recall data from the first preset specified when the fade amount is greater than or equal to the threshold, and will recall data from the second preset specified when the fade amount is less than the threshold. e.g. inverted thresh: 'fade' < thresh = value b; fade >= thresh = value a;
  - exponential curve: Power curve. Takes an additional argument (float), which sets the exponent to which the fade amount will be raised. Presets recalled will recall data between the two specified presets, along the curve described. Power curves can be used to create faster or slower "attacks" and "decays" for the fade envelope;
  - table: Table-specified curve. Takes optional additional argument (see [transition curve](#) below), which specifies the name of a table to use for curve lookup (tab1, tab2, tab3 or tab4). Presets recalled will recall data between the two specified presets, along the curve described in the table. Tables are assumed to contain values between 0 and 100, representing the new fade amount \* 100. If the lookup fade amount does not fall exactly onto a table-specified value, linear interpolation is used to determine the new fade amount. In Density.m4l you have four draw functions to draw your curve, see [transition curve](#) below.

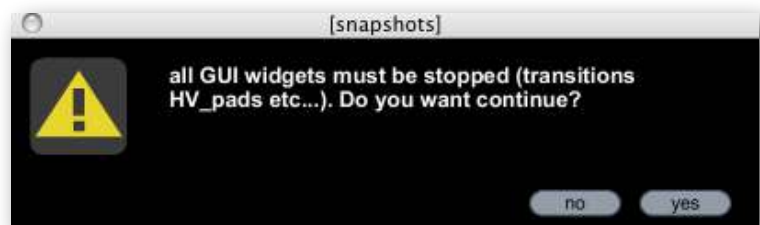
The storage window displays any stored presets. The active (recalled) preset is displayed in highlight green. If any client value is changed, are displayed in italics. Eventually, both of these windows will be configurable and editable, so that they can provide display and editing control for clients and storage sets.



Name	1	2	3	4	[99]: de...	100: GU...
pitch	-24.	-21.732...	-24.	-24.	0.	-24.
volume	-6.	-6.	-6.	-6.0002	-6.	-6.0002
pan	0.	0.	0.	0.	0.	0.
rnd_pitch	0.	0.	0.	0.	0.	0.
jitter	0.	0.025974	609.611...	0.060998	2.	0.
scann	0.042638	0.042838	0.042734	0.042838	1.	0.042838
distance	1.	1.	1.	1.	0.5	1.
pan-freq	1.	1.0002	1.	1.0002	1.	1.0002
rnd_dens	13.3858...	13.3860...	13.3845...	13.3860...	33.	13.3860...
rnd_len	14.9606...	14.9608...	14.9592...	14.9608...	33.	14.9608...
density	173.042...	6.987041	31.111677	659.645...	100.	173.042...
length	210.975...	868.072...	269.918...	190.387...	85.	206.983...
scrub	6091.69...	18453.9...	5536.54...	19172.9...	0.	24547.6...

N.B. When you open storage window, a message box will be displayed:

this is to avoid an overhead of CPU when displaying storage window.



## transition curve

All transition by defaults are linearly interpolation. You can superimpose your interpolation curve at one or more subscribed widgets. Four draw functions are available, from toolbar or menu bar. Open one of them and draw your curve. This feature is then saved in the Density device.

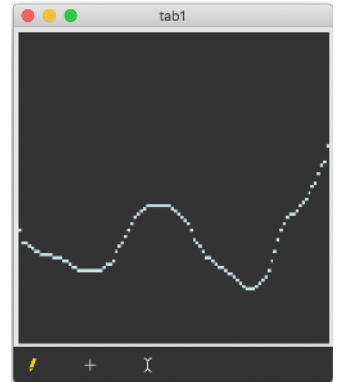
Draw a shape on the function window

In order to maps one or more subscribed clients with one of four functions draw, you need open clients window and select table from pop-up menu (over interp header). Now you must digit the table name, that can be: tab1, tab2, tab3 or tab4.

in the above example, we employ:

- tab1 like transition curve for density, rnd\_density, rnd\_length;
- tab2 like transition curve for scrubb;
- pan, pitch, rnd\_pitch and scann are unsubscribed from the snapshots transitions system.

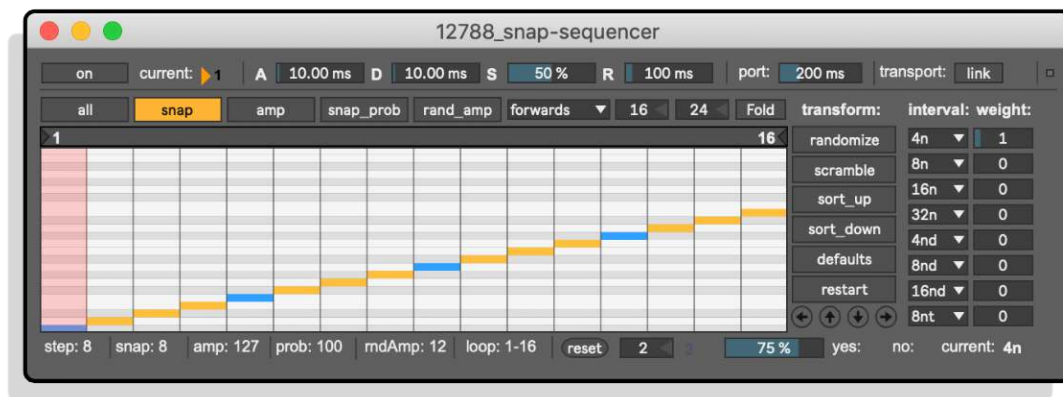
See client window above for more explanations.



Name	Priority	Interp		Data
<input checked="" type="checkbox"/> backward	0	threshold	⇅ 0.50	0.
<input checked="" type="checkbox"/> density	0	table	⇅ tab1	1.
<input checked="" type="checkbox"/> distance	0	exponential c...	⇅ 2.00	0.590551
<input checked="" type="checkbox"/> jitter	0	exponential c...	⇅ 0.50	648.818909
<input checked="" type="checkbox"/> length	0	inverted thres...	⇅ 0.00	63.960632
<input type="checkbox"/> pan	0	linear	⇅	192.
<input type="checkbox"/> pitch	0	linear	⇅	-9.543307
<input checked="" type="checkbox"/> rnd_dens	0	table	⇅ tab1	33.
<input checked="" type="checkbox"/> rnd_len	0	table	⇅ tab1	15.748032
<input type="checkbox"/> rnd_pitch	0	linear	⇅	0.
<input type="checkbox"/> scann	0	linear	⇅	1.
<input checked="" type="checkbox"/> scrub	0	table	⇅ tab2	141.732285
<input checked="" type="checkbox"/> volume	0	linear	⇅	-27.26



## Snapshots sequencer improviser unit



Snapshots sequencer rhythm improviser unit i.e. the extended name, the aim of this device is to control the snapshot sequence88 by improvising with different rhythmic pulses. The sequencer shown on vertical the snapshots numbers while in horizontal steps sequence.

### beats cycle

A metronome is synced to Ableton Live global transport time and outputs a trigger on each beat. The beats are counted in what we're defining as a cycle of  $<n>$  beats (Cycle sub-patcher). Each time the cycle restarts, we're choosing whether or not we want to change the current time interval. This is set by using a probability factor (ChangeProb sub-patcher). If the program decides to change the time interval, a trig is sent.

### time intervals

Since eight time intervals can be selected. The choice is performed with a random function, according to a weight probability for each possible interval - an interval with a weight of 2 will have twice as many chances to be selected by the random procedure that an interval that has a weight of 1.

### step sequencer

When a given time interval is chosen, it is used as a step rate for driving the step-sequencer. The step-sequencer is synced to Ableton Live global transport time, so the steps are played in accordance with transport tempo. The snap-sequencer has many editing modes that allow for modifying the snapshots content (snap, amp, etc..).

### output

The snap-sequencer outputs a list of data each time a step is triggered. Each data type (snap, amp, amp\_rand...) is interpreted in order to recall a snapshot, with additional possibilities such as: random amplitude of envelope and snap recall probability.

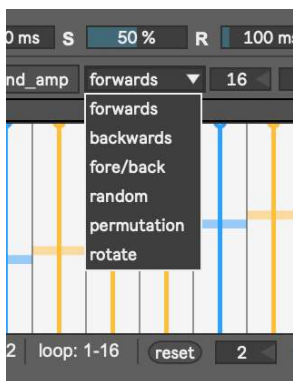
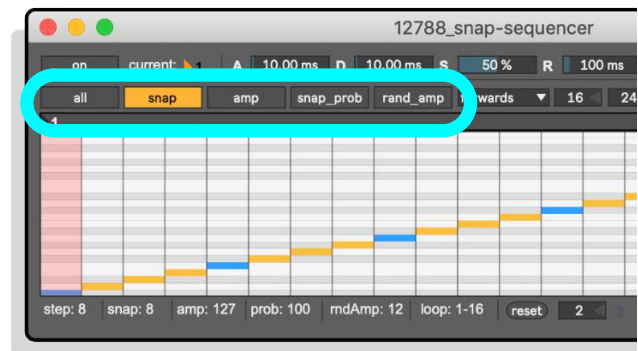
Up to eight time intervals can be selected. The choice is performed with a random function, according to a weight probability for each possible interval - an interval with a weight of 2 will have

twice as many chances to be selected by the random procedure that an interval that has a weight of 1.

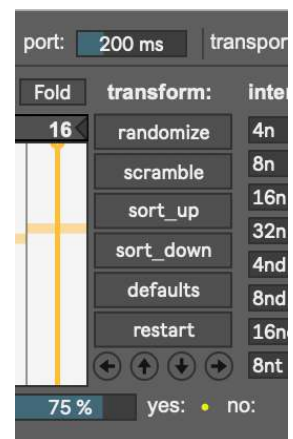
You can set "general changes probability" (50 % in the example) , 0 % no interval change occur, as all intervals probability weight to 0. Change and no change led (yes/no), shows if the random parts it's to change or not. Random number generator fall every "change interval" time (2 in the example), beat sequence is displayed on the right.

The most left reset button, resets sequencer to first step, in permutation mode reinit sequence, no action in random mode.

Edit mode selection is done using this set of tabs, view all displays all of the available data (and allows you to edit the snapshots). Snap mode displays and allows editing of the snap only. Amp mode allows you to edit the volume of snapshot envelope. Duration mode is used to change the step size of each value, while Probability mode allows you to manipulate the likelihood of a snapshot.



Sequencer play mode menù read steps in the explained way. You can chose the number of the step (columns) and snapshots range (row), then fold/unfold lets you chose to display all possible snapshots, or only a specific set of snapshots.



Some kinds of transformations are available their names are auto-explicative.

Through the four arrow keys at the bottom, up and down (snap number) or left and right (in time) you can shift the entire sequence.



Snapshots Sequencer is an ADSR (Attack Decay Sustain Release) every step envelopes the streams (or main) amplitude audio signal. You can define the Attack & Decay times along with the Sustain percentage and finish with the Release time.

When the Sequencer is turned off, audio signals (streams or main) pass through normally.



## Note and Tick Value

Here is a listing of the note and tick values associated with common note durations. Note value abbreviations that can be used in Density to specify time are in bold.

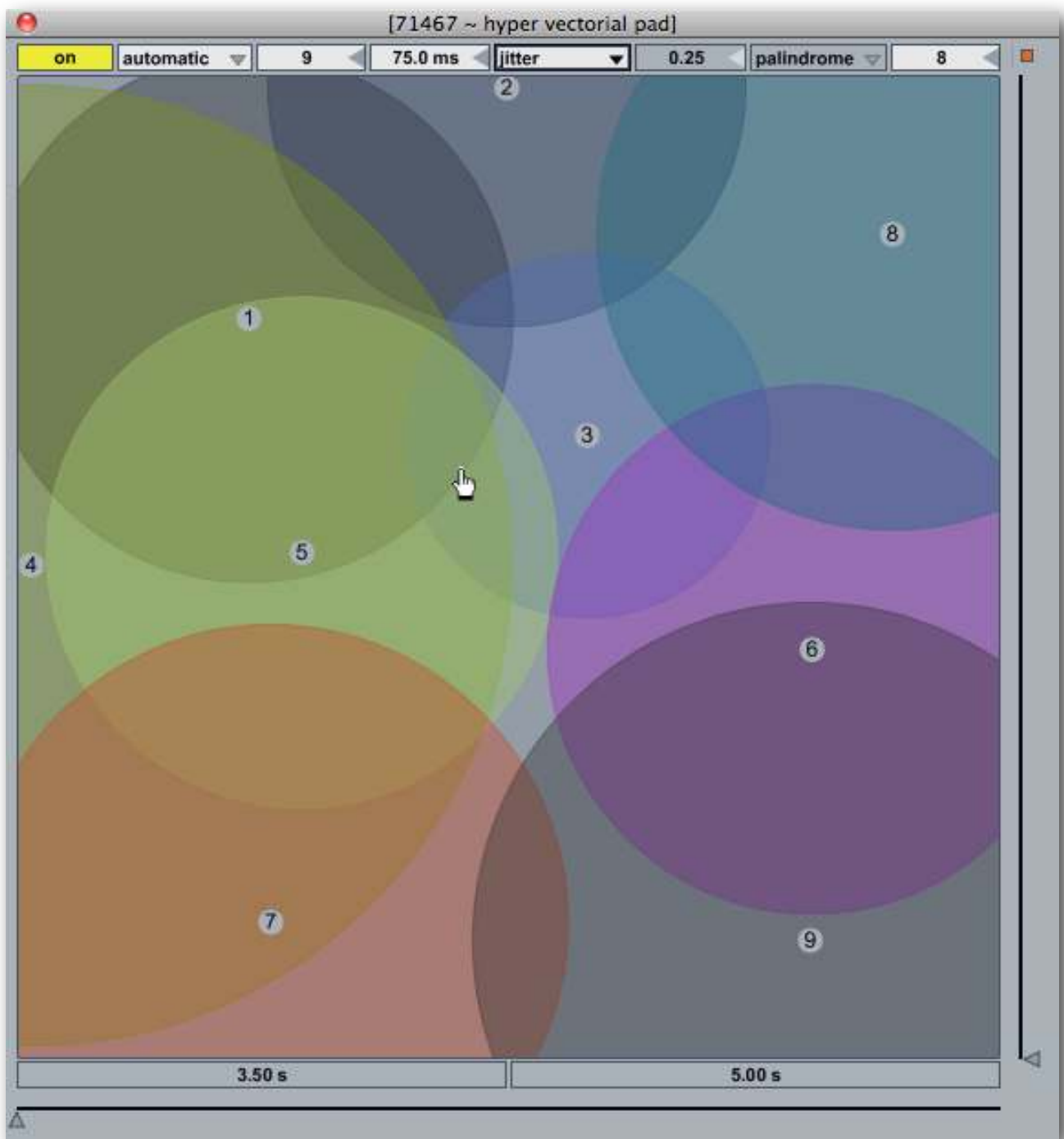
128n - One-hundred-twenty-eighth note - 15 ticks  
 64n - Sixty-fourth note - 30 ticks  
 64nd - Dotted sixty-fourth note - 45 ticks  
 32nt - thirty-second-note triplet - 40 ticks  
 32n - thirty-second note - 60 ticks  
 32nd - Dotted thirty-second note - 90 ticks  
 16nt - Sixteenth note triplet - 80 ticks  
 16n - Sixteenth note - 120 ticks  
 16nd - Dotted sixteenth note - 180 ticks  
 8nt - Eighth note triplet - 160 ticks  
 8n - Eighth note - 240 ticks  
 8nd - Dotted eighth note - 360 ticks  
 4nt - Quarter note triplet - 320 ticks  
 4n - Quarter note - 480 ticks  
 4nd - Dotted quarter note - 720 ticks  
 2nt - Half note triplet - 640 ticks  
 2n - Half note - 960 ticks  
 2nd - Dotted half note - 1440 ticks  
 1nt - Whole note triplet - 1280 ticks  
 1n - Whole note - 1920 ticks

1/128  
 1/64  
 1/64D  
 1/32T  
 1/32  
 1/32D  
 1/16T  
 1/16  
 1/16D  
 1/8T  
 1/8  
 1/8D  
 1/4T  
 ✓ 1/4  
 1/4D  
 1/2T  
 1/2  
 1/2D  
 1/1T  
 1/1

In Density you can find also this formula of value.

## Hyper Vectorial Pad<sup>1</sup>

### general explanation

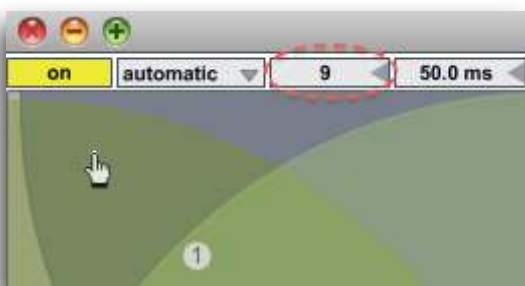


HV\_pad (i.e. Hyper Vector Pad) is 2D parameter interpolation user interface. Displays nodes in a 2-dimensional space, and calculates the distance from a point. The distance factor determines the weight between snapshots interpolation. Like for snapshots-module, we have ten HV\_Pad grouped in a unique window.

You can open HV\_Pad window from the tool bar button.

<sup>1</sup> See VMCI Virtual Midi Control Interface (<http://www.csounds.com/maldonado/>), thanks Gabriel Maldonado.

HV\_Pad extends the functions of micro-pad, interpolating until 24 snapshots together, thus you can modify hundreds of parameters with a single mouse motion, according to a structure configured by the user. This control method is called Hyper Vectorial Synthesis Control. Normally the user interacts with these areas with the mouse or through self-scanning exploration, however a remote MIDI control is possible, by assigning the X Y pointers (see [Ableton Live user manual](#)).



In the image below, from number box (red circle) you can set the number of nodes, you can set a specific configuration (system nodes) by clicking and dragging center of node to move it or Option Key (alt) to change node size.

N.B. The nodes are numbered progressively, therefore each node (on the pad) must correspond by a stored preset snapshot. If you employ an empty snapshot, the interpolation between them will not work properly. Also you cannot disabling completely the nodes (node area = 0), to avoid fall in the same problem.

## pad mouse behavior



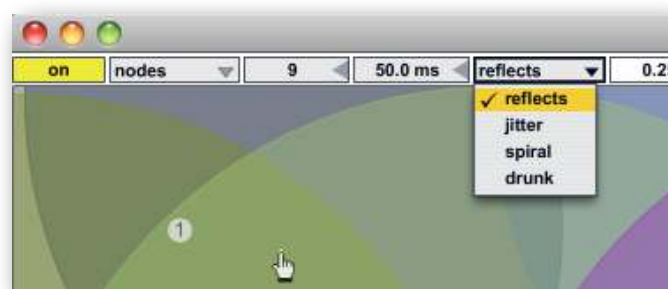
The pop-up menu allow you to chose mouse behavior:

**Automatic:** allows the mouse to change the nodes position, size and slider (pad scrub);

**Nodes:** the mouse can only edit the nodes position and size, alt + mouse scroll to change node size; click and drags on the node number, to move it;

**Slider;** the mouse can only changes the slider location (pad scrub).

You can automatize HV\_pads scrub (slider) choosing a scanning mode:



## Scanning method:

**reflects** causes a reflection when the cursor reaches the edge;

**jitter** outputting random numbers within a moving specified range around current scrub mouse position;

**drunk** will perform a "drunk" walk, creating unpredictable paths;  
spiral centripetal, centrifugal or palindrome, with phase deformation parameter;

The jitter step size (number box 8) it will be showed only when jitter is selected. This parameter sets the random amount of jittering around of the mouse position during scrub pad.

Automatic exploration in reflects, jitter and drunk, you can interact with the pad. The values of mouse and direction, will be the new coordinates of scanning. Instead in spiral you cannot interact with pad

## X/Y time

The X Y indexes scanning ratio, depends by the "frequency" X/Y axes. These are controlled by the two sliders showing value in seconds (i.e. time = 1/frequency).



You can set scanning time for the two axis (2D XY). For example 3.50 sec. for the X axe it means that horizontal slider scrub it take 3.50 sec to pass from left edge to right edge of the pad, Y = 5.00 sec.

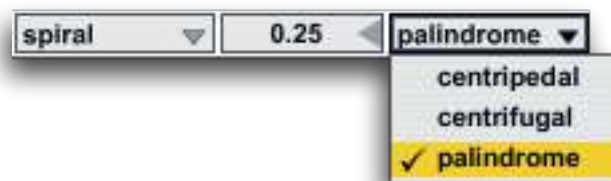
X Y parameters, they assumes different means according to selected scanning mode:

in reflects is expressed in time, is the time it takes to move from X = left/right and Y = up/down;

in drunk is expressed in time, is the deviation amount  $\text{rnd}(X)$  and  $\text{rnd}(Y)$ ;

in spiral: X is the time for each sub-spiral, Y is the total duration until reach all pad area;

The spiral mode, will enable two related parameters:



centripetal moves from outside to inside, when it reaches the limit again, repeat;

centrifugal does the opposite;

palindrome is a combination of both;

The spiral phase deformation (number box 0.25) 0: no spiral; 0.25 circular; > 0.25 crushed; > 0.5 clockwise rotation

## miscellaneous

Auto-exploration is possible only when the Ableton Live transport is enabled (see [Ableton Live user manual](#)).



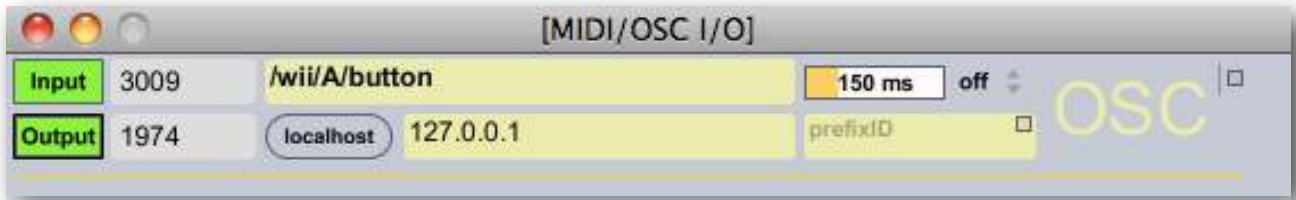
on/off toggle auto-exploration according to selected mode.

The red circled number box, sets the interpolation update time in milliseconds, this is a speed limit to avoid an overhead of cpu with complex interpolations, be careful with short times!

## OSC I/O

### header

Open OSC I/O from [quick access](#) buttons, you can configure the OSC (Open Sound Control) protocol, through the header,



### OSC:<sup>1</sup>

Dispatch messages through an OpenSound Control address hierarchy with pattern matching. Values are serialized and sent over the network as OSC compatible UDP;

**Input port** (3009) set input port to send messages to at host;

**Output port** (1974) set output port to send messages;

**Green left buttons**, enable/disable respective OSC I/O.

**Very important!: default values for I/O ports: 30-09-1974 are the author birthday date.**

We need also configure host (127.0.0.1) output , entering an IP address or hostname, clicking on localhost to assign default ip value (i.e. local host 127.0.0.1).

OSC monitor show OSC I/O data, you can select with mouse the text string address (OSC path) and paste it on widget controller (see below).

You can enter a prefix ID (can be numbers and/or text) for the the OSC output address. When an incoming OSC message is synced on a widget parameter, this read and write values employing the same path address. Some time is useful append an id prefix to differentiate path in order (for instance) to send message to another device .

N.B. OSCulator <sup>2</sup> can be employed like MIDI/OSC monitor beyond supplying advanced features.

N.B. All settings will be saved on the project (see [overview](#) for more explanations)

<sup>1</sup> OSC is a CNMAT Max objects, can be found at: <http://www.cnmatic.berkeley.edu/MAX>

<sup>2</sup> OSCulator is a software that can be used with many different hardware devices and software. For example, with OSCulator, your Nintendo Wiimote can talk to major MIDI sequencers or your favorite console emulator or even the Kyma sound design workstation. And your [iPad](#), [iPhone](#), [iPod Touch](#) or [Lemur](#) can do just the same... with great ease of use. <http://www.osculator.net>

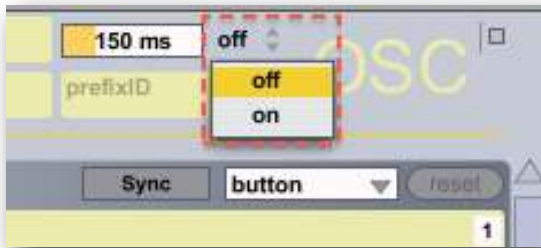
## sync

Sync system allow you synchronize software parameters with hardware devices. Thus Density.m4l send out through OSC ports parameters data. Default sync is disabled for all widgets, then no OSC data are sent. You can select enable sync out for each widget controller or superimpose a common feature at all widgets (see below).

For example, you can enable sync for all widgets controllers, by selecting the item from header menù.

Sync have five outputs mode:

OSC out mode for all widgets:



**off** no OSC outs data;

**on** send OSC data continuously. N.B. this overwrite all settings of the widgets.

Every time you move a Density.m4l parameter on the GUI or when a transition/sequencer/HV\_pad is at work, sync values are sent.

To avoid an overhead of outputs data, through the number box (150 ms) you can change MIDI/ OSC output update rate in milliseconds. For instance during the snapshots transitions, you should limiting sync data in order to preserve a normal cpu overhead: OSC out update rate, expressed in milliseconds set the speed limit OSC output data.

- widget mapping

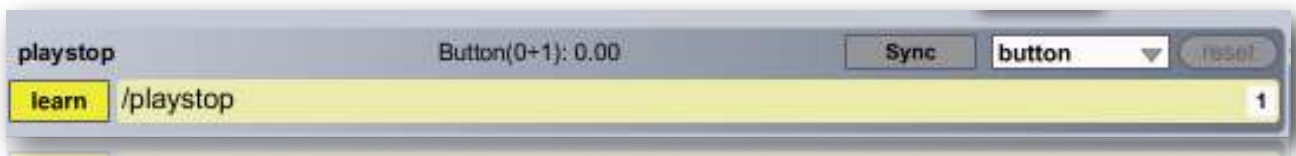
We have two widgets mapping kinds:

1. Cc\_raw
2. Cc\_scale

Cc\_raw is addressed to mapping Density.m4l buttons/toggle parameters, Cc\_raw read/write OSC returning integer values (between 0 and 127), Learn function waits to feel an OSC address, when it happens the the control is set up on the OSC path address.

Now you can controlling Density.m4l parameter via OSC.

Of course you can manually change OSC address.



Toggle to learn and send an OSC data in order to assign incoming OSC address. In the example the OSC /play/stop address are assigned to play/stop Density.m4l parameter, if sync is enabled Density.m4l also is able to send OSC data. It is possible change OSC path value entering a new one.





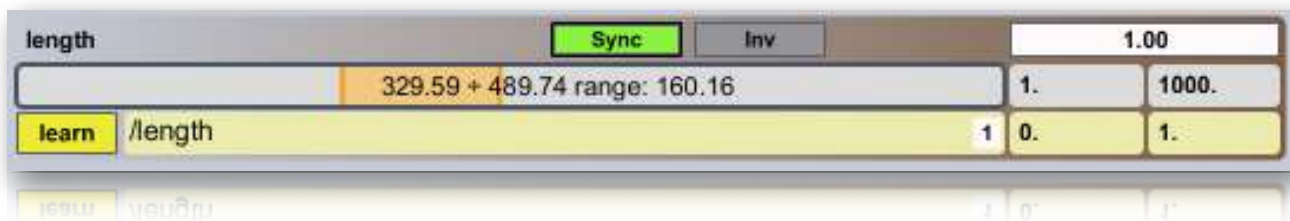
An OSC message can be a list, if you want read an element of the list, you can select an OSC input item from the list, (default first element). In the example, Nintendo Wii device send the accelerometer values like a three list items, selecting 3 to read only third incoming.

I/O sync pop-up menù, set output mode for current widget, see [sync](#) for better understand.

The most right menù (toggle), set the controller kind:

- toggle switch from 1 and 0 every time MIDI/OSC I/O == 0;
- button send 1 every time MIDI/OSC input != 0;
- incr and decr increment or decrement (regardless of direction) by one step, every time MIDI/OSC input == 0. when values 127, or 0 are reached, the next value is wrapped;
- incr/decr it counts upward until it reaches 127, then counts down until it reaches 0, then up, then down, and so on;
- controller mode read raw data from MIDI/OSC input, raw data MIDI between 0 and 127.

Cc\_scale extends Cc\_raw functions, you can choose a certain range on which rescale the parameter, for example control length, which is associated, covers a range -70 ÷ +6 (76 units).



In the example, incoming OSC data is scaled on a part of the entire parameter range by the orange horizontal selection. You can also also change min/max values for the parameter (box numbers -70. and 6.), current range will be reported.

OSC assumes default values between 0 and 1. You can also change OSC input range. For example we control gain1 through /wii/1/accel/roll OSC data incoming from OSCulator and Nintendo Wii, we rescale the range for controlling only a middle part of parameter.

Inv, invert the I/O range for the widget.

Most right menù, I/O sync (off) set the output mode for current widget, see [sync](#) for better understand and

numbox (1.00) on top right set the exponential base value (default 1. = linear). The number is converted according to the following expression:  $y = b e^{-a \log c} \text{ ex } \log c$ .



## Overview

### status bar

In the Density status bar, are shown sound files information and grains overlaps factor.

On the left we have grain overlaps factor, you need enable tiny orange button. Follows WavePad sampled file informations like: current file name, current file length in sample, current file sampling rate, current file length in milliseconds.

### info window

To avoid annoying hint messages when you are working on the GUI, Density sends all hints in the Ableton Live Info window. In the clue window you can read some informations about Density GUI widget, when the mouse is over it. See [Ableton Live user manual](#).

### float window

Have you noticed all Density windows contains in a corner (generally top right) a tiny orange button? toggle it to set float/no float window. When toggle float, current window will be always in front. This feature is saved on project like general setup.

## acknowledgments

I would like to thanks Eugenio Giordani and Nicola Casetta, a special thanks to: Renato Alberti, Felix Petrescu and Pasquale Ascione.



CNMAT ©

OpenSoundControl (OSC-route)

<http://www.cnmat.berkeley.edu/MAX>

All rights reserved

Isotonik Studios would like to thank

Chaos Culture & Tom Cosm

The Author:

Alessandro Petrolati

apeSoft

All rights reserved © 2010-2019

[info@apesoft.it](mailto:info@apesoft.it)

[www.apesoft.it](http://www.apesoft.it)