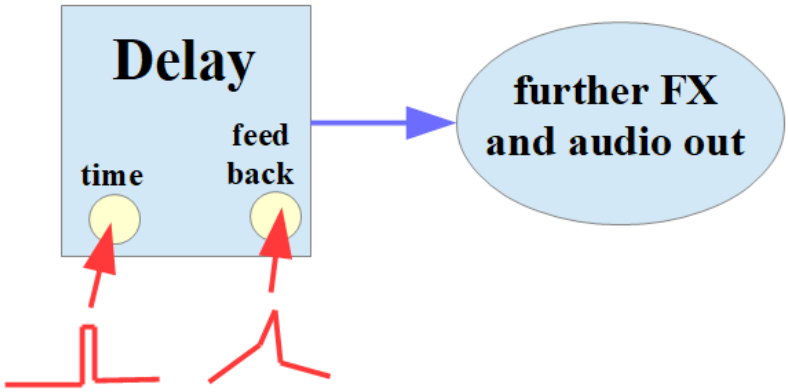
Welcome to article 14 about making generative music with modular synths (more about this matter at https://dev.rofilm-media.net)

Chapter 2.5:
Effects

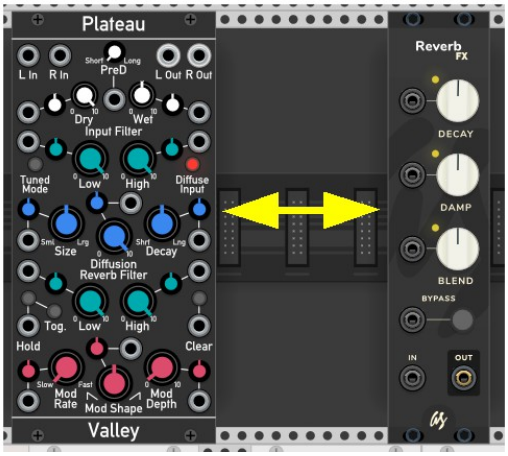
The number of effect modules is tremendous, and the number of all of their parameters is nearly uncountably high.

And all of them can be targets for modulation and randomisation. Let me choose only 2 examples from this group of targets therefore: the probably most often used delay and reverb effects.

Delay first. When we randomly modulate the delay time with short pulses while e.g. a melody is fed into the module, we get short bursts of kind of alien-like sounds, which set (random) accents here and there. And we add even more changes, more development to the piece, when we modulate the feedback with any other wave shape than pulses or squares, and let both modulations run at the same time, but at different frequencies (wave lengths). A simple pitch development (melody) can became acceptably interesting this way. The preset “delay_1.vcv” (downloadable presets are only in the e-book – see https://dev.rofilm-media.net) and the video behind the following link demonstrate this.
https://youtu.be/qAtVcHFsldI



Next example: Reverb
There are quite simple reverb modules with only a view parameters (e.g. only the decay time and the HP-damp), as well as extremely versatile and multifunctional delay modules with tons of reverb parameters.



I use only the decay time in this next (and last in this sub-chapter) example, because all of you will have this parameter on their reverb modules.
Modulating the decay time leads to an effect like reverb echoes. Not the reverb generating sound itself is echoed, but only its reverb (we get a "corrugated reverb tail). The preset "reverb_1.vcv" and the video behind the following link demonstrate this.
https://youtu.be/hq8Wk34SoY0

Chapter 2.6:
Envelopes

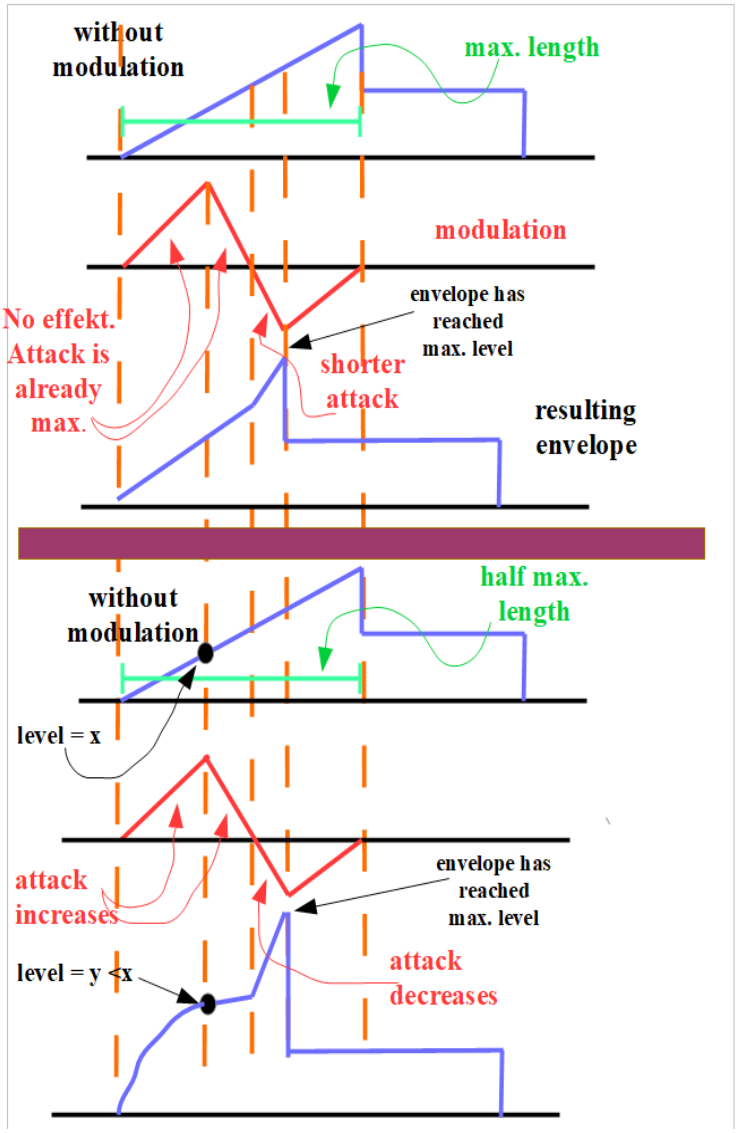
It´s necessary to talk about randomising envelope parameters a bit more in detail now. In chapter 1 we saw that envelopes can be a modulation **source** as well, that they can be patched generating cycles of more or less complex structure.
Now let´s talk about envelope parameters as modulation/randomisation **target**.

Modulating the attack time, the decay time or the release time leads to rather subtle than dramatic changes.
Why?
Well, it´s all about time, and nobody and nothing can go back in time (at least not in "normal life – some discoveries in quantum physics seem to show, that under certain circumstances …). Time that has gone by is "lost" for ever.
So, when I speed up the attack time by the falling flank of a triangle wave (just an example) I always shorten the whole attack time. No matter if the rising flank arrives early enough to slow down the **remaining** "head

space” again.
(don´t forget: a falling CV level means **less** attack time = faster, a rising CV level means **more** attack time= slower.)

AND:
Once having reached the maximal output level of the module the envelope will inevitably jump to its next phase, the decay time (or stay at or fall to sustain level, if decay is adjusted to zero).

The following graphic shows an example.
We see: modulating the attack time will **always** shorten the overall attack time compared to the time, which the attack phase would last without modulation.



And it´s the same with the other time domain parameters decay and release.
The preset “envelopes_1.vcv” and the video behind the following link will help you understanding.
https://youtu.be/yMM6nEjOhmQ

When the gain signal and the modulating signal are out of phase, then we get different attack times (or different decay times or different release times) with each new gain, but they all will be shorter than the unmodulated version.
And that is why attack, decay and release modulations are rather subtle (shorter effects) than dramatic most of the times.
The presets “envelopes_2.vcv” and “envelope_3.vcv” and the videos behind the two following links demonstrate some low and medium complex patch examples using envelope modulations.
https://youtu.be/sNq71Z71K7E

https://youtu.be/D_k1RdxOwbg

We have already seen in chapter 1, how to randomise/modulate regular or random cycles producing modules, and that we can modulate/randomise mixer channels shouldn´t be a surprise either. No extra sub-chapter needed, I think.

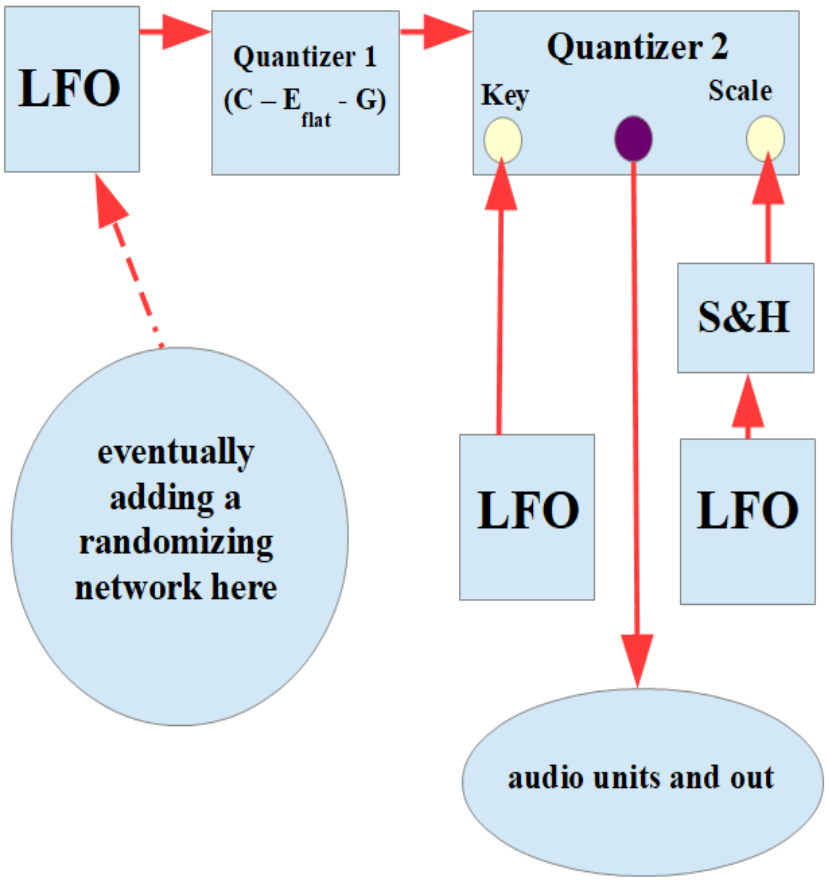
Chapter 2.7:
Quantizers

A lot of quantizers offer CV input jacks for modulating (randomizing) the scale and the key of a series of incoming pitch CVs. When we do so, it is (most of the times) a good idea to completely randomize only one of these parameters, and modulate the other one by a (more or less complex) repeating regular cycle of CVs.

And why not feeding this quantizer by another one, which produces only a very few different pitches, e.g. allowing only the notes of a certain chord. This way we make changing the key and the scale (one of which randomly changing) better recognisable even for “ordinary” listeners.

When the unit(s), which deliver(s) the pitch CV for this non-transposing quantizer is a regularly repeating cycle (e.g. an LFO or a network of

LFOs), then a relation of integer multiples between the three participating frequencies (see graphic) lets the piece sound nearly kind of “classic with a disturbing touch”.

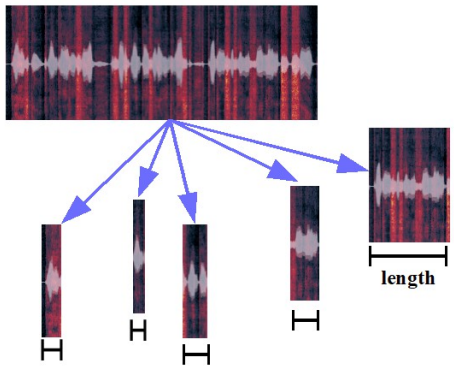


The preset “quant_1.vcv” and the video behind the following link show an example and may serve as a”base camp” for experiments of your own.
https://youtu.be/d0xv6axbnrE

Chapter 2.8:
Grains

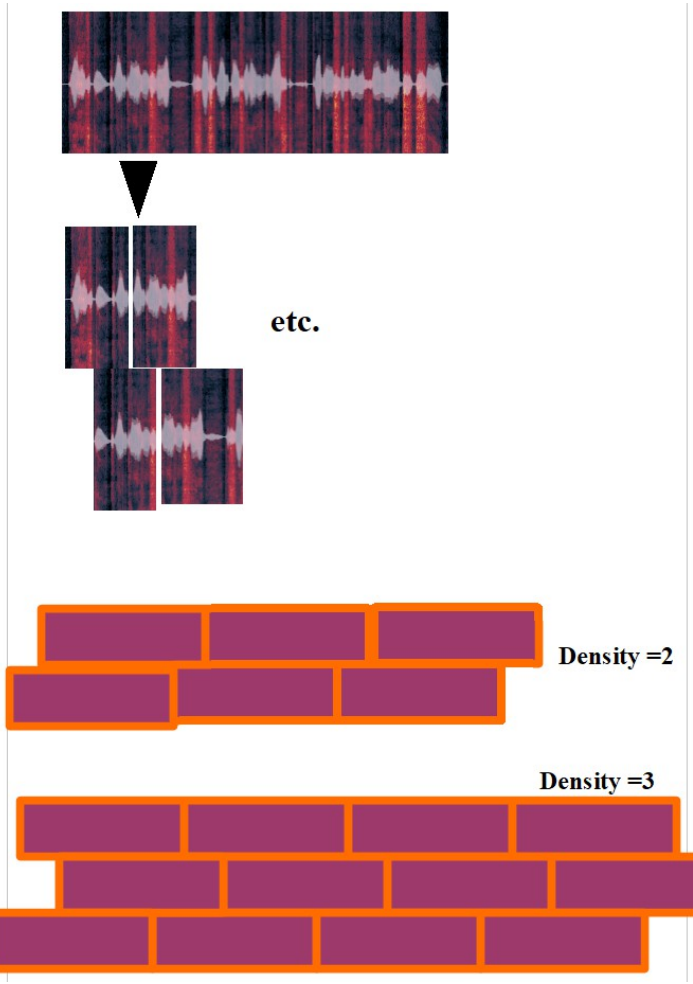
Granular sound processing is probably the widest field we touch in this book. I´ve written a whole book of more than 270 pages about the matter of sonic grains, and it´s only Volume 1 of at least 3 (2 more to come). I´ve produced a series of about 20 videos concerning granular sound design (some of which are hours long).
A grain and a stream of grains can have dozens if not hundreds of different parameters, and each manufacturer emphasizes on different one. But there are at least three parameters which we can consider basic (and which can be found with (nearly) all modules): the length of a grain, the density of grain streams and the pitch of a grain. The direction a grain is played back

in is a major parameter as well, but not all modules offer the possibility to modulate playback direction.
Anyway, in the modular world there are not **that** many granular sound processors (opposed to the world of software instruments, where there are a lot of them). Let me focus on the three main and major and most basic parameters therefore. And for those of you, who still need at least a basic understanding of how grains work the following short and quite simple explanations may come in handy.



A sonic grain is basically a (very short) snippet of audio. The length of a grain can reach from only a millisecond (1/1000 second) up to some minutes (60 seconds is a common upper border with a lot of modules). Each grain can be pitched differently (and to different tones than the original audio). The pitch of a granular processed piece of audio is independent from its length/playback speed (the explanation how this is achieved shall be left to another book).

How many (overlapping) grains of the same piece of audio are played back at the same time is described by the “density” parameter. Important to know: grains, which are sounding at the same time don´t need to represent exactly the same piece of audio.



If we have a piece of audio, which is remarkably longer than the average grain length, the position from where a grain is taken is important as well.

There is a huge difference whether a module can work on audio, that is streaming in/through the module (real time granular processing), or if we have to load a certain piece of audio in the module, where it then can undergo our granular processings.

Alright! Enough of theory. Let´s jump into granular tweaking now. I have chosen the (very good software reproduction) of “Clouds” by Mutable Instruments. And even if I´m not going to explain the whole module (I´ve produced a video doing so – it´s nearly 2 hours long!), I´ll demonstrate the parameters length, density, pitch and position and there modulation using this module.

A last warning: Because granular processing intervenes deep into the heart of a sound, it is easy to get dramatic sonic results – but it is not at all easy to get musically meaningful/useful results.

The presets “grain_1.vcv” and “grain_2.vcv” and the video behind the following link may help you doing some steps of your own.
https://youtu.be/LV_vqYCEWBY

Chapter 2.9:
Sample (Player)

Modulating and randomising parameters like loop-start, loop-end, playback direction etc. with samplers and sample players is kind of general knowledge I think.

The preset “sampler_1.vcv” and the video behind the following link demonstrate one way to apply sample players to our generative music setup:

The main sounds generating process is interrupted and a pre-recorded sample is played back.
The interruption of and the returning to the main sounds is executed by a Gray Code module, and loop start and loop end of the sample player are modulated by an LFO, so that we don´t always hear the same part of the sample, when our nicely ambient sounds are “brutally hushed”.
https://youtu.be/18_LrErjIxw

… to be continued