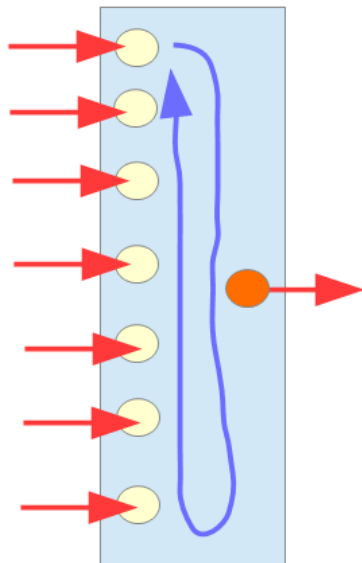


Chapter 1.2.4: Sequential Switches

We were standing in the “garden of sources of regularly repeating CV” and were looking over the fence into the realm of randomness in the last chapter, which I could have called “Changing Randomness into Regularity”.

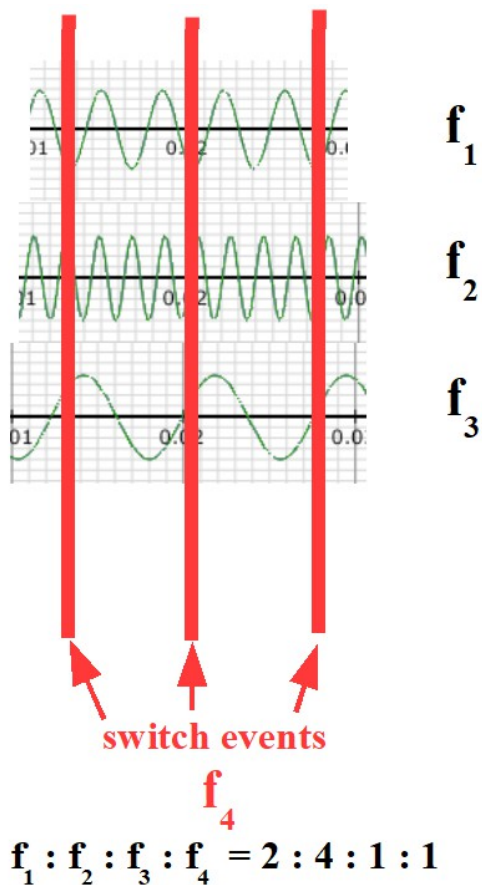
In this chapter we are standing in the same garden again, but at the opposite side and we are going to look over another fence into another realm, the realm of switches, which are no sources at all – neither of CV, nor of anything else. And again I’m going to “misuse” them and I’m going to force them into pseudo sources of complex but regular and regularly repeating CV cycles.

There will be a whole chapter about switches later in this book, but let’s deal only with one certain kind of (sequential) switches now, which takes a number of different inputs and leads each of these inputs sequentially to one output.

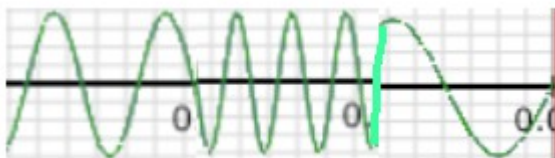


And now I take as many CV wave shapes – e.g. from LFOs – as the switch has inputs and patch each of them to one of the inputs. What I get at the output is a regularly repeating complex wave shape (of CV), which consists of all constituent individual shapes. The result may be a cycle consisting of a sine followed by a square followed by a triangle followed by a saw etc. and this again and again. Or I take more complex shapes to the inputs – e.g. the results of a couple of modulation networks.

The only thing I have to take care of (if I want something regular) is that the frequency of change between the inputs as well as the frequency of all input generating units are integer multiples of each other. Otherwise we would lose our wanted regularity.



The resulting (and regularly repeating) CV development generated by the situation, that is depicted in the graphic (switching from f_1 to f_2 to f_3 back to f_1 etc.) would look like this:



(It's not necessary to have the same wave shapes!)

With non-integer frequency relations each switch would “catch” the input shape at always different phases leading to an always changing and never repeating succession of resulting wave shapes.

But wait a minute! Do we really lose regularity? Well, we do not, of course. We simply get what we got at the very beginning of this chapter 1: we combine a couple of waves, which are running at different frequencies

and different phase shifts, which means an actual situation will surely occur again after some time, and from then on repeat again – only that this “some time” can be a quite long one, which causes the **impression** of randomness. And again its on us, it depends on our compositional decision how much regularity and how much randomness we want at a certain point in our patch.

The preset “sequentialSwitch.vcv” (only in the ebook, see <https://dev.rofilm-media.net>) and the video behind the following link may lead you deeper into this matter.

<https://youtu.be/9h0Ug8yY9Ao>

Chapter 1.2.5: The Turing Machine – Part 1

In the chapter “Generative Potential of Certain Modules” I’m going to introduce the Turing Machine and its applications in generative music in detail. But there is one aspect, one function witch makes it seem reasonable to mention this module even in this early chapter here.

In chapter 1.2.3 about shift registers we met the situation, that on one hand we didn’t know which sequence of CV levels we would get, but – once got – this development of CV levels would regularly repeat over and over again. With the Turing Machine we can create a similar situation (and therefore we are looking over the fence into the realm of randomness again).

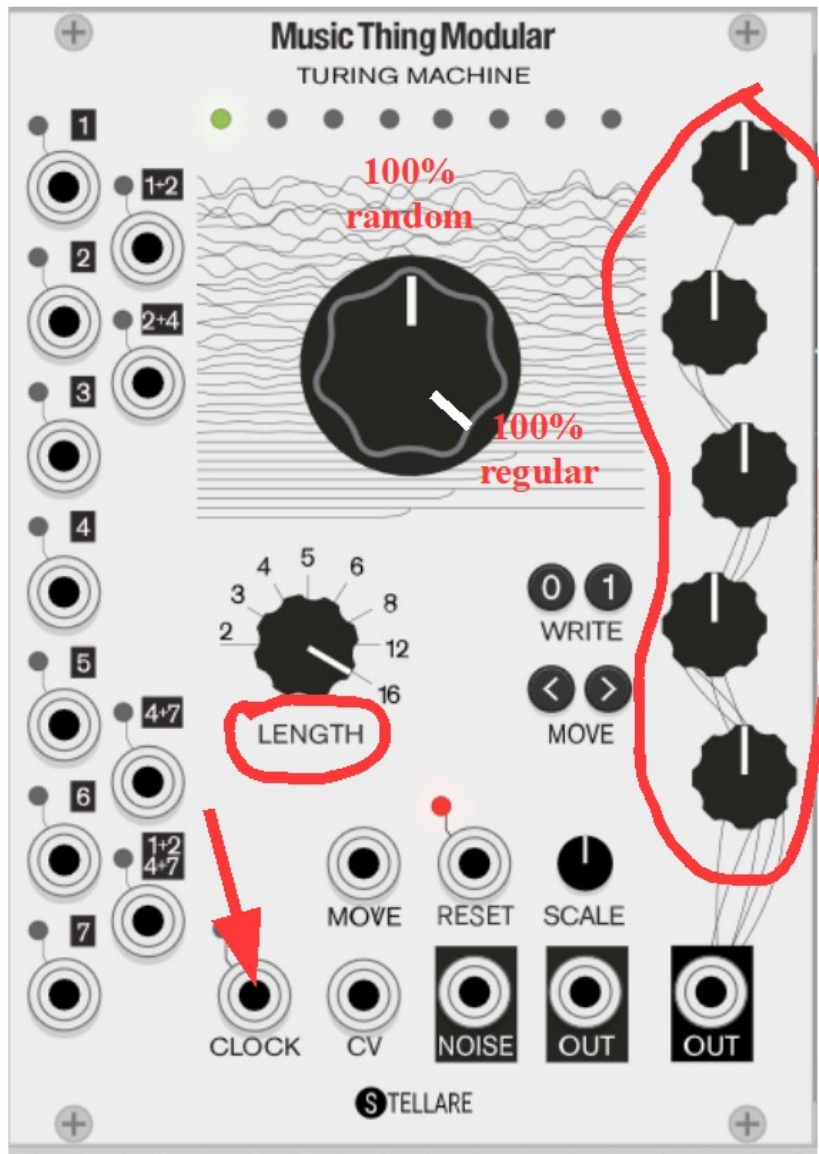
Let me make it short and simple here. The big knob in the upper center of the picture sets the amount of randomness the Turing machine will produce. In it’s 12 o’clock position we get only and 100% random successions of CV levels, whereas in the 5 o’clock position the sequence/succession of CV levels, which is in the buffer of the module at a certain moment won’t change any more – we have “caught” or frozen the sequence, which will regularly repeat as long as we sustain the five o’clock position of the knob.

The smaller knob below and a bit to the left sets the number of steps of the sequence/succession of CV levels in both situations: in randomness as well as in regularity. We can say it sets the length of the sequence.

And the five knobs at the right let us adjust certain steps manually, which

means we can change the CV level of certain steps of the sequence, **which is in the buffer of the module at the moment.**

And the speed/rate is set by the CLOCK input.



The preset “turing_1.vcv” (only in the ebook, see <https://dev.rofilm-media.net>) and the video behind the following link show the way to your first experiences with the module. To read about its whole functionality, please go to chapter 5 of this book.

https://youtu.be/wj-Z_SCKzgw

Chapter 1.2.6: Samples and Recordings

And why not recording/sampling CV, and play it back in loop mode? It

would be a regular CV development, a regular and repeating cycle of CV levels, right?

And what's more, these sampled CV cycles – which may be the result of a complex network of LFOs, sequencers, shift registers etc. - can easily be stored away and used anywhere and at any time as completely independent units without the need of loading all the modules or even the whole patch, which generated these CV cycles some time ago.

And the functionality of a sampler with its loop-start and loop-end and other functions can change us into a cosmetic surgeon of CV cycles.

The preset “sample.vcv” (only in the ebook, see <https://dev.rofilm-media.net>) and the video behind the following link introduce you to the matter.

<https://youtu.be/JlrTSzfbXk>

And combining all the mentioned ways of generating regularly repeating cycles of CV levels we can build really complex and long cycles, and generate quite perfect illusions of “ever changing” events.



... to be continued