

Fortuna, a Semi-Random Generator

Vittorio Tondo

MA Commercial Music Producer

CPL0921F2

Development Module, Element 2b

Word Count: 3641

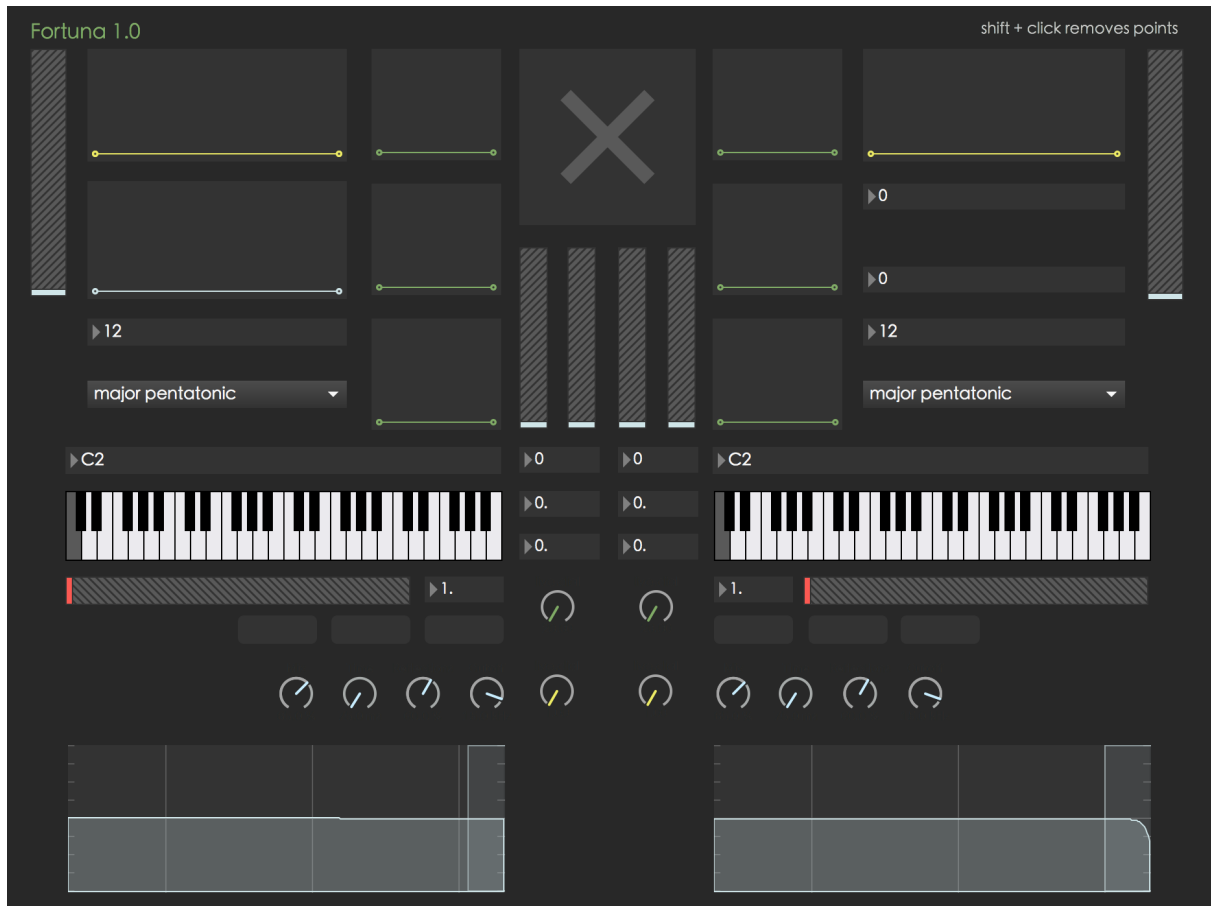
In this essay, I will illustrate the research and methodology used to create the instrument, describe the basic principles applied to design it, and the motivations that led me to lean toward this type of synthesiser. I will outline the issues I encountered during development and how I circumvented them. Furthermore, I will compare the instrument with other similar devices on the market, discuss features and issues that came up in beta testing, and outline a roadmap for improvements to come with future updates.

The instrument is a Max/MSP patch. It is a semi-random composite synthesiser with two sections: a Frequency Modulator and an Additive synthesiser. The FM synthesiser has two sine wave oscillators, whereas the Additive synthesiser uses three sine waves and three saw waves.

The two synthesisers are controlled by two independent, semi-randomised MIDI sequencers and two scale and root note quantisers, which snap the random frequency output from the sequencer to the closest note. The speed at which notes get generated is random, but the user can roughly control it with two millisecond-scaled controls.

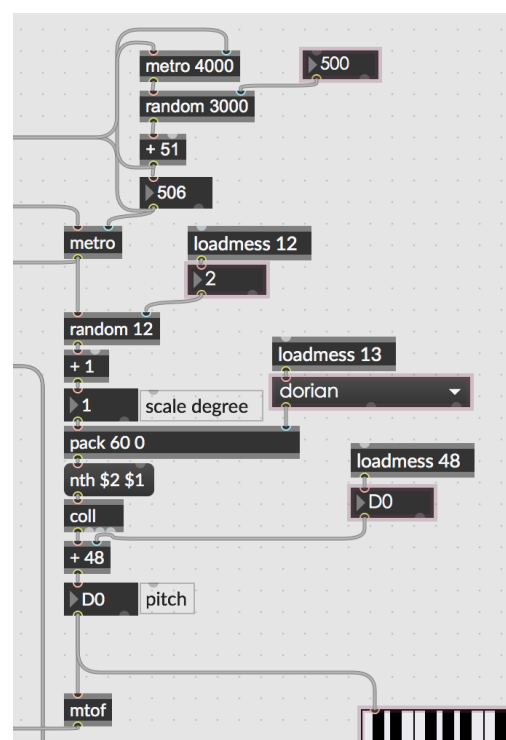
Each synth has a master envelope control that has the potential for virtually infinite envelope points. The FM synth has a modulation envelope control, while the resonant oscillators of the additive synth have each an independent envelope control.

Both sections also have a controllable filter with lowpass, highpass, and bandpass options, a delay with vibrato, a parallel overdrive, and a reverb section.



Fortuna's interface

The instrument starts by activating the sequencers via the central x-shaped button. The button sends a “bang” to a random time generator (controlled by the objects “metro” and “random”), which the user has some control of. The patch generates a number from 1 to 12 at a relative random time. The number then gets quantised to a scale mainly through the use of a “coll” object, which is a matrix of scale degrees, informed by the user-controlled mode menu (“umenu” object). An



integer/MIDI object then sets the scale's root note; the user has control over this function.

1	0,	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
2	1,	0	2	4	5	7	9	11	12	14	16	17	19	21	23	24	26	28	29	31	33	35	36	
3	2,	0	2	3	5	7	9	10	12	14	15	17	19	21	22	24	26	27	29	31	33	34	36	
4	3,	0	1	3	5	7	8	10	12	13	15	17	19	20	22	24	25	27	29	31	32	34	36	
5	4,	0	2	4	6	7	9	11	12	14	16	18	19	21	23	24	26	28	30	31	33	35	36	
6	5,	0	2	4	5	7	9	10	12	14	16	17	19	21	22	24	26	28	29	31	33	34	36	
7	6,	0	2	3	5	7	8	10	12	14	15	17	19	20	22	24	26	27	29	31	32	34	36	
8	7,	0	1	3	5	6	8	10	12	13	15	17	18	20	22	24	25	27	29	30	32	34	36	
9	8,	0	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40	4	
10	9,	0	3	7	10	14	17	21	24	27	31	34	38	41	45	48	51	55	58	62;				
11	10,	0	3	6	9	12	15	18	21	24	27	30	33	36	39	42	45	48	51	54	57	60		
12	11,	0	2	3	5	6	8	9	11	12	14	15	17	18	20	21	23	24	26	27	29	30	32	
13	12,	0	1	3	4	6	7	9	10	12	13	15	16	18	19	21	22	24	25	27	28	30	31	
14	13,	0	2	4	7	9	12	14	16	19	21	24	26	28	31	33	36	38	40	43	45	48		
15	14,	0	3	5	7	10	12	15	17	19	22	24	27	29	31	34	36	39	41	43	46	48		
16	15,	0	12	24	36	48;																		

Coll object mode matrix

MIDI, standing for “Musical Instrument Digital Interface”, is a universal technology standard that allows instruments, processors, and computers to communicate with each other. Practically all digital musical devices have been using MIDI since 1982. (Rothstein, 1922)

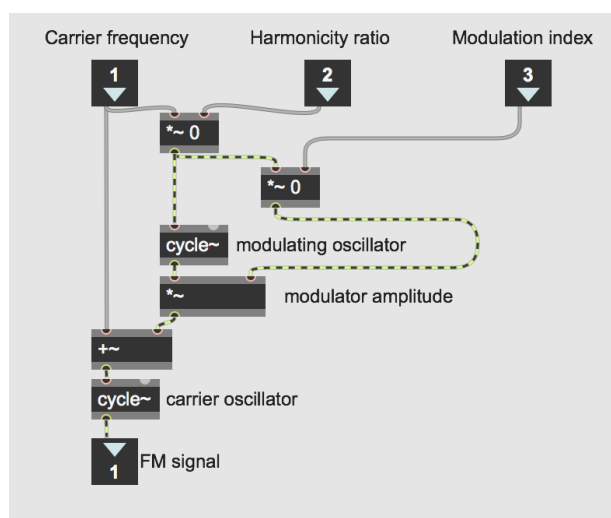
The sequencer described above is a tempo randomiser/quantiser. Quantisation is “the process of converting a continuous range of values into a finite range of discreet values” (Media College, 2020). Although being used in other fields, such as electrical engineering and mathematics, this process has become ubiquitous in music production since the advent of the first digital sequencers.

The signal is then routed into a scale quantiser, with the mode and root note selected by the user.

Finally, the MIDI signal generated is transformed into an audio frequency integer by the object “mtof” and sent to the respective sequencer’s synthesiser.

FM synthesis is a type of audio synthesis that creates complex waveforms via the modulation of one carrier frequency by the amplitude of a second modulator frequency (Charles and Jerse Thomas A, 1997). As previously mentioned, the first signal is sent to a Frequency Modulation synthesiser. Although the process was once applied to radio transmission, it was only in 1977 that John Chowning applied the model to audio spectra (Chowning, 1977). Then in 1983, the first mass-produced FM synth came on the market, the Yamaha DX7, which changed the musical landscape forever (Yamaha Corporation).

To program an FM synth, I have used two sine waves (object “cycle~”), with the modulator’s harmonicity ratio and modulation index being randomly selected and not controllable by the user; this gives the synth an unpredictable and ever-changing quality. The modulation index is also controlled by using a function editor, which allows the user to draw with their mouse the change in modulation over time.

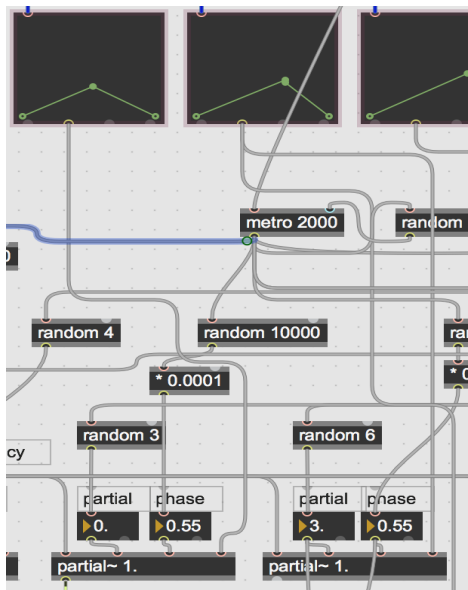


Simple FM synthesis in Max MSP

The second synthesiser uses additive synthesis; it creates timbres and tones by adding up simple soundwaves, based on Bernoulli's discovery that all sounds occurring in nature are composed of several summated sine waves (Darrigol, 2007) (de Poli, 1983). A sine wave is a "waveform that represents periodic oscillations in which the amplitude of displacement at each point is proportional to the sine of the phase angle of the displacement, and that is visualised as a sine curve" (Merriam Webster, 2019) when translated into sound, this wave has a clear, pure-sounding tone.

I combined six oscillators to achieve additive synthesis, alternating three sine waves and three saw waves (the fundamental being a sine wave). A saw wave is a non-sinusoidal waveform that ramps upwards in amplitude and then drops sharply (Black, 2022), and a richer sound characterises it. As previously described, all sounds are composed of sinewaves, which is also valid for sawtooth waves.

Each oscillator has four controls: the fundamental frequency, a frequency ratio about the fundamental, a phase control, and amplitude control. Both the ratio to the fundamental and the phase controls are entirely randomised. The amplitude control of each oscillator is controlled independently by a different function-based ADSR envelope and the synth's global envelope control.



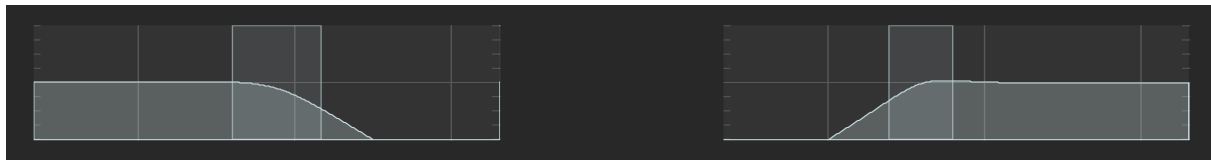
Additive synthesiser

An envelope is a control that defines the movement of a given parameter over time. ADSR stands for Attack, Decay, Sustain and Release, and it represents the change in amplitude of a sound over time (Music Tech Student, 2017). For example, the sound of a piano generally has a fast attack time, a rapid decay, a sustain determined by how long the musician presses the key, and a release time determined by the sustain pedal being held down. The modulation function editor for the FM section above is also an envelope, as it controls modulation over time.

Each oscillator forming the additive synth has its phase modified randomly over a randomly generated period of time. “Phase denotes a particular point in the cycle of a waveform, measured as an angle in degrees or radians” (Indiana University, 2019). Phase is not generally of consequence when dealing with a single waveform. Still, it is majorly important when dealing with multiple waveforms, especially when talking about sine waves, since if “one wave is at its maximum amplitude while the other is at its minimum amplitude,

the sound waves are said to be out of phase.” (Knowlton, 2017); this means they cancel each other out and result in silence, a phenomenon called “Destructive Interference.”

Both signals are routed into independent filters, which the user can set to one of three modes. An audio filter is a frequency reactive circuit that can increase or attenuate a specific frequency, or above or below it, within a range of 0Hz to 20kHz (Kalnaskas, 2022). The cutoff control sets the frequency that the filter has to work from; the resonance sets either the boost or the reduction of the given frequency.



Lowpass and Highpass filters

The audio signal gets split into three equal signals (for both synthesisers): one going to the output, one into an overdrive, and one into a delay section.

Guitarists first used distortion towards the end of the 1940s as they discovered that turning up valve amplifiers beyond “safe” levels would generate a crunchier, warmer tone (Polyphonic, 2018). Distortion works by increasing the amplitude of a soundwave beyond an amplifier’s maximum capacity. The wave gets “clipped”, generating a compressed tone and harmonic artefacts in the higher frequencies. (Davis and Jones p. 102 201, 2000).

Delay is an effect that plays back a stored audio input after a given delay time, usually expressed in milliseconds. Avantgarde musicians such as Pierre Schaeffer created the first delay effects using magnetic tapes during the 1940s (Justin Guitar, 2021). Fortuna’s delay

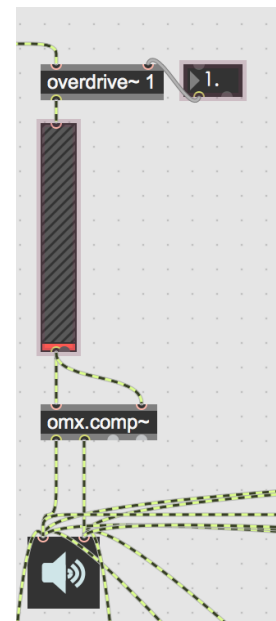
also has a vibrato effect, which fluctuates the pitch of the wet signal by a regular pulsating amount, controlled by a depth and tare amount, controllable by the user.

The overdrive and delay are used as “parallel effects”, as they are routed separately from the dry signal; this gives the user the possibility of mixing and matching the dry and wet signal, offering more sonic design opportunities.

The overdrive signal does enrich the sound; however, the level of the signal was pretty unmanageable at higher levels, so I decided to run it through a compressor by using the function “omx.compression~”, as illustrated on the right side.

A compressor is “a variable gain device, where the amount of gain used depends on the input level. [...] the gain will be reduced when the signal level is high, which makes louder passages softer, reducing the dynamic range” (Lehman, 2003). It was first used on the radio during the 1930s - 1940s to ensure the sound of the presenter talking and the audience clapping would be of similar amplitude. It has since then become ubiquitous in music and used in virtually all commercial productions (Musich Tech Student, 2013).

The delay signal is then routed into a digital reverb effect, which simulates how sound acts in real-world locations, like rooms or halls. Sound bounces in all directions before reaching our ears when emitted from a source, giving it a sense of space, and digital reverberators emulate the effect by using a series of filters and delay effects (Lehman, 2003). The reverb has four user-controlled parameters: “wet”, controlling the amount of reverb,



“time”, controlling the length of the effect, “reflections”, controlling the number of simulated reverberations, and cutoff, as the reverberations get routed through a lowpass filter, to give the illusion of being further away; this is because lower frequency waves travel further than higher frequency waves, as they conserve energy for a more extended time (Physics Lab, 2022), as well as the Doppler effect, whereby audio sources moving away from the listener sound lower-pitched (The physics classroom, 2019).

The objective I was setting out to achieve when starting to design Fortuna was twofold. Firstly, I wanted to develop a synthesiser that would be unconventional, with the distinctive characteristic of not having on-screen feedback allowing the user to identify the function of the instrument’s controls. The objective of this quirkiness is to stimulate creativity through constraint. As described by Candy (2007), “constraints within a particular [musical] genre are changed over time until an entirely new genre emerges” however, this hypothesis applies not only to music but to all forms of creativity. World history is full of examples in which necessity became the mother of invention: for example, the look of the popular character Super Mario was dictated by processing limitations (McLaughlin, 2010). Secondly, I am curious about aleatoric music, or music composed partly or entirely by chance. From composers such as John Cage, who wrote “Music of Changes” by following the Chinese divinatory system named “I Ching”, to Iannis Xenakis’ “Pithoprakta”, based on “mathematical theories of probability related to the motion of gases” (Malinverno, 2021), randomness is an ever-providing source of creativity. I set out to create a program that could make generation as easy and fun as possible.

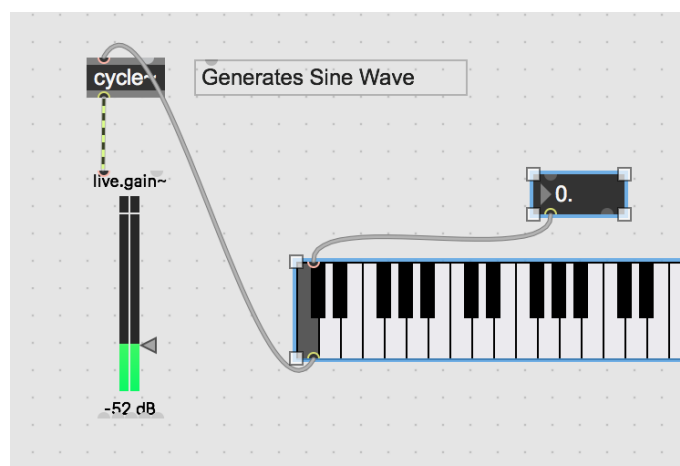
The decision I took to include an FM synthesiser derives from the fact that, by nature, it tends to be a more unpredictable form of synthesis, therefore keeping in line with the wish

to create unexpected results. Furthermore, FM is more likely to develop sounds further away from the natural world than other synthesis techniques (Hahn, 2020).

In contrast, Additive Synthesis is more predictable and generates sounds more closely related to real-world instruments, such as organs. I picked these two methods to have two different sides of the coin and ensure more sonic variety (Reid, 2020).

Creating the instrument has been a somewhat challenging process, as I have never had any experience using any kind of coding language and have relatively limited synthesis knowledge. As a first task, I set out to generate a simple sinewave, which seemed pretty straightforward in theory, but I could not get any sound from the patch.

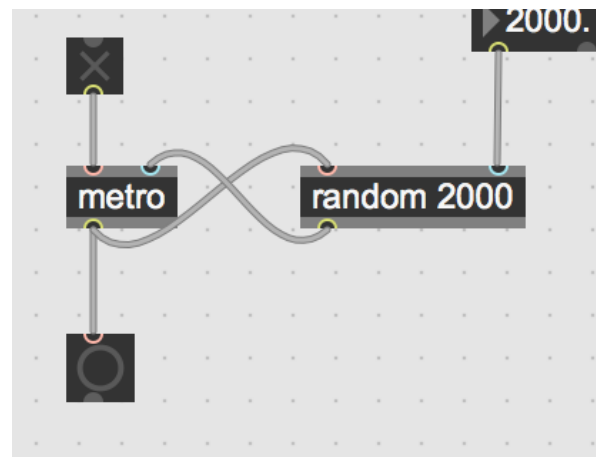
I had not realised that I had to change the digital audio generated from the `cycle~` object into an analogue signal using a “`dac~`” object (B, 2020).



The next step was to generate a random time and note value to be routed into the oscillator. Since Max has a “random” object, it seemed the process would be simple. However, “metro” objects generate output signals, or “bangs”, at a predetermined interval. In contrast, “random” objects need to receive an input (as previously mentioned, a “bang”) to generate a random value. It was, therefore, challenging to create a random chain of events if the main bang was sent by a non-random source regularly. I overcame this by “looping” a metro object with no set value with a “random” object, creating a random loop.

The loop became one of the fundamental connections behind the patch, which I used extensively. (Robinson, 2021)

I encountered the next roadblock when looking for a way to quantise the randomly-generated frequency value to a given scale - this was necessary as I thought that even a random generator would sound too



alien if it were too far from equal temperament. After trying to use integer objects set to “MIDI” to convert the signal to a note approximation, I came across a tutorial describing the process of using “coll” objects (as illustrated above) to convert numbers into scale degrees (Berklee Online, 2022).

Next, I had to convert the pitch data from MIDI into a frequency to be fed to an oscillator. I spent about half a day looking for a solution to the issue, and, while discussing the problem with a fellow student, I was made aware of the “mtof” object, a simple function that took care of the task perfectly.

To generate more exciting sounds and improve the instrument’s versatility, I decided to add a reverb to the effect chain, locating the device in a digitally simulated space. I followed instructions to create a reverb from scratch using the gen~ object to replicate a mathematical reverberation model developed by Manfred Schroeder and Ben Logan in the early 1960s (Stanford University, 2010). However, after following the tutorial closely, I still could not identify a noticeable change in the final output. I, therefore, decided to reverse

engineer one of the pre-programmed reverbs present in Max/MSP, which took me a lot less time and was way more effective.

I decided to follow a minimalistic design in terms of the user interface. The reason for following this direction is twofold. The device's lack of numerical feedback or labelling can confuse a new user (as shown in beta-testing); I, therefore, intended to avoid creating an even more complicated experience by using a more elaborate design scheme. Secondly, minimalism is often a testament to the quality of the product presented, as underlined by C.S. Jones (Jones, 2015)

The primary colour used for the instrument's design is a muted shade of black. I decided on this particular colour because of its association with darkness, the night and the unknown. My intention is for a user to go into a new patch without knowing what to expect, being open to being inspired and embracing randomness as a creative force.

Lastly, I named Fortuna after the Roman goddess of Luck, which seems a fitting name for a generative synthesiser.

The closest instruments on the market at the moment, and a source of inspiration for my device, are modular synthesisers.

As defined by thomann.de: "Simply put, a modular synthesizer is an electronic musical instrument that consists of a multitude of different components (modules) that are used and combined to create electronic sounds. The individual modules are connected via cables (patch cables), switches, sliders and patch panels." (Joe, 2019).

Modular synths have been around since the 70s (notable players being Moog and Buchla); however, they have recently seen a massive resurgence in use, as all analogue gear has in the last few years, due to characteristics such as tonal quality and resale value (H. Jones, 2019).

Although modular systems are more versatile and powerful than Fortuna, they come at a distinctly higher price point. Where Fortuna will be marketed as a “Pay What You Like” (PWYL) software, the price for a single oscillator module ranges from roughly £70 to £700 (Thomann); this puts Fortuna at a net advantage in comparison.

Another potential competitor to Fortuna would be VCV Rack, a modular synthesis software emulator developed by VCV. It works the same way as a modular system and has two versions: a free and a pro version. Although the free version has a good number of modules available, some modules have to be bought, while others are only accessible with the pro version of the program. Furthermore, my intention with Fortuna is to develop an instrument that will give possibilities not necessarily achievable through a modular system (for example, a hundred-oscillators additive synthesiser.)

Lastly, a “random generator” bundle is currently on offer by Audiomodern. It is a collection of five plugins that offer user-controlled random generation of chord progressions, melodies, beats, filters and gates. Although the bundle is marketed as an aleatoric generator, the user controls it mainly, which contrasts with what Fortuna sets out to do: not what the user wants. Furthermore, the Audiomodern generators are not part of the same instrument, but they come with a price tag of £109 (Plugin Boutique, 2022). Because of these reasons, I think my device has an edge.

The competition's products differ from Fortuna in one crucial element: they offer visual feedback and direction for the user: I honestly think the lack of this sets my device apart.

Two participants took part in beta-testing. Both have extensive experience with synthesis, and one of them owns a modular system.

Participant A was intrigued by the instrument but mentioned two caveats: Firstly, he thought the lack of a noise generator was prominent, as he felt it would give a further dimension and thickness to the sounds generated. Noise is used in synthesis to simulate sounds such as wind and breaths (White, 1994), and it would be an exciting addition to Fortuna. As defined by Jeffrey Stolet: "noise generators output all frequencies distributed randomly through the entire audible spectrum" (Stolet, 2022).

Secondly, the participant thought it would be helpful to give the user the possibility of writing their notes on each section of the synthesiser, both "gamifying" the user experience and stimulating a learning process related to synthesis.

Participant B found great enjoyment in pushing the boundaries of what is achievable with the synthesiser, even managing to simulate a noise generator by raising the tremolo rate past audible rates. He mentioned it would be helpful to add a "clear all" function to start a fresh patch instead of having to turn the controls back to zero manually. Furthermore, he suggested UI and ease of use improvements, such as giving the user the ability to change the root note by pressing it on a graphic keyboard instead of manually typing it. Also, he

suggested the timing controls be changed from millisecond controls to bpm controls, making implementation into DAWs more seamless.

The latter is easily achievable using a simple equation: $60000 / \text{ms} = \text{bpm}$ (Guitar Gear Finder, 2020).

All the suggestions from the testers have been very enlightening, and I have added them to my improvement schedule for future updates. Furthermore, I will constantly have other musicians test different device iterations to receive improvement ideas that I might not have considered myself.

While using Fortuna, I have been keeping track of what I consider are features I would like to develop and prioritise on top of the ones suggested by beta testers.

An element that I felt was lacking from my instrument but present in almost all synthesisers is a Low Frequency Oscillator (LFO). An LFO is a time-variant controller that oscillates at sub-audio frequencies (usually below 20Hz) and can be routed to control one or more parameters of a synthesiser (Stolet, 2019). LFOs are used to automate the evolution of a sound over time, making it more exciting and often more organic; adding it to Fortuna with some random parameters (e.g. modulation speed) would be an excellent opportunity to generate a more varied and exciting sound.

Furthermore, I feel the addition of a random sampler could be beneficial in offering a more diverse soundscape. Freesound.com has made an API available to download sounds from their website into a max/MSP patch; with an extra layer of randomness, it would be

possible to have a section on the synth that randomly loads unknown samples and plays them back to the user, offering an infinite soundscape (felicss, 2010).

I also feel a music generator would not be complete without a rhythm section; therefore, I plan to add a semi-regular, synthesised drum machine to Fortuna soon. It is crucial, though, for it to be somewhat aligned with the other elements of the sequence, as there is a fine line between complete random and controlled randomness; humans find comfort in repetition in music (O'Bannon, 2022)

Lastly, I feel it is necessary to offer some more experimental options that the competition is not providing. As mentioned previously, the idea to create an additive synthesiser composed of a hundred or more random oscillators sounds very striking to me as it would be both intriguing to complete and offer something not currently available on the market.

In conclusion, I am delighted with the final result of my instrument. Although it has been a difficult journey, I have learned a lot about audio programming and synthesis.

User feedback has been overwhelmingly positive regarding the device's design and capacity to stimulate creativity and enjoyment. I, personally, have found myself spending much time playing around with it with glee.

Although Fortuna is far from finished, I have a clear roadmap ahead of me, one that will see me improving its current features and adding new ones.

One thing is for sure: this will not be the last instrument I will create.

Works Cited

- B, M. "Max/Msp - First Patch (Generating Sine Wave) - Tutorial 1." *W*www.youtube.com, 2 Nov. 2020, www.youtube.com/watch?v=hfa-XxyvoC0&ab_channel=MB. Accessed 9 Mar. 2022.
- Baudoux, Roald. "Converting an Integer into a Midi-Message - MaxMSP Forum | Cycling '74." *Cycling74.com*, 12 Sept. 2008, cycling74.com/forums/converting-an-integer-into-a-midi-message. Accessed 10 Mar. 2022.
- Berklee Online. "Max MSP Tutorial -- the Absolute Basics Part 3 -- Helpful Objects to Know." *W*www.youtube.com, 1 Feb. 2021, www.youtube.com/watch?v=V2SCRTij1sU&ab_channel=AndrewRobinson. Accessed 10 Mar. 2022.
- Black, Ken. "What Is a Sawtooth Wave?" *W*www.wise-Geek.com, 5 Mar. 2022, www.wise-geek.com/what-is-a-sawtooth-wave.htm.
- Candy, Linda. "Constraints and Creativity in the Digital Arts." *Leonardo*, vol. 40, no. 4, 2007, pp. 366–367, www.jstor.org/stable/20206448. Accessed 31 Mar. 2022.
- Charles, Dodge, and Jerse Thomas A. *Computer Music : Synthesis, Composition, and Performance*. S.L., S.N, 1997, p. 115.
- CHOWNING, JOHN M. "The Synthesis of Complex Audio Spectra by Means of Frequency Modulation." *Computer Music Journal*, vol. 1, no. 2, 1977, pp. 46–54, www.jstor.org/stable/23320142. Accessed 30 Mar. 2022.
- Darrigol, Olivier. "The Acoustic Origins of Harmonic Analysis." *Archive for History of Exact Sciences*, vol. 61, no. 4, 6 Apr. 2007, pp. 343–424, 10.1007/s00407-007-0003-9.
- Davis, Gary, and Ralph Jones. *The Sound Reinforcement Handbook*. Milwaukee, Usa, Hal Leonard Corporation, 2000, pp. 102–201.

de Poli, Giovanni. "A Tutorial on Digital Sound Synthesis Techniques." *Computer Music Journal*, vol. 7, no. 4, 1983, p. 8, 10.2307/3679529. Accessed 11 May 2020.

felicss. "FreeSound.org Hack for Max/Msp/Jitter 5." *Felicss's Blog*, 8 Nov. 2010, felicss.wordpress.com/2010/11/08/freesound-org-api-hack-for-maxmsp-jitter-5/. Accessed 1 Apr. 2022.

Guitar Gear Finder. "How to Convert Ms (Milliseconds) to BPM (Beats per Minute) and Vice Versa." *Guitar Gear Finder*, 23 Mar. 2020, guitargearfinder.com/guides/convert-ms-milliseconds-bpm-beats-per-minute-vice-versa/. Accessed 1 Apr. 2022.

Hahn, Michael. "FM Synthesis: Everything You Need to Know to Get Started." *LANDR Blog*, 30 Mar. 2020, blog.landr.com/fm-synthesis/.

Indiana University. "Acoustics Chapter One: Phase." *Cmtext.indiana.edu*, 2019, cmtext.indiana.edu/acoustics/chapter1_phase.php.

Joe. "Modular Synthesizers - a Simple Explanation." *T.blog*, 29 Sept. 2019, www.thomann.de/blog/en/modular-synthesizers-explained-simply/.

Jones, C. S. "Minimalism vs. Ornament: Is Either More "Relevant?"" *Creative Market Blog*, 1 July 2015, creativemarket.com/blog/minimalism-vs-ornament-is-either-more-relevant. Accessed 31 Mar. 2022.

Jones, Hollin. "5 Reasons to Use Analog Gear in Your Studio." *Ask.audio*, 5 Aug. 2019, ask.audio/articles/5-reasons-to-use-analog-gear-in-your-studio.

Justin Guitar. "Delay: What, How & History." *Justinguitar - Free Guitar Lessons with Justin Sandercoe*, 2021, www.justinguitar.com/guitar-lessons/echo-tape-delay-analog-delay-and-digital-delay-fx-301.

Kalnoskas, Aimee. "Basics of Audio Filters." *Analog IC Tips*, 24 Feb. 2021,
www.analogictips.com/basics-of-audio-filters/. Accessed 31 Mar. 2022.

Knowlton, Chris. "Introduction to Phase." *Discovery of Sound in the Sea*, 27 July 2017,
dosits.org/science/advanced-topics/phase/.

Lehman, Scott. "Harmony Central®: Effects Explained: Compression/Limiting."
Web.archive.org, 12 Apr. 2003,
web.archive.org/web/20030412155119/www.harmony-central.com/Effects/Articles/Compression/. Accessed 31 Mar. 2022.

---. "Harmony Central®: Effects Explained: Reverberation." *Web.archive.org*, 2 Apr. 2003,
web.archive.org/web/20030402010546/www.harmony-central.com/Effects/Articles/Reverb/. Accessed 31 Mar. 2022.

Malinverno, Matteo. "The History of Randomness and Chance in Music - Blog | Splice."
Www.splice.com, 29 Mar. 2021, splice.com/blog/randomness-chance-music-history/.

McKloskey, Bendan. "Random between Two Numbers - MaxMSP Forum | Cycling '74."
Cycling74.com, 16 Apr. 2012, cycling74.com/forums/random-between-two-numbers.
 Accessed 9 Mar. 2022.

McLaughlin, Rus. "IGN Presents: The History of Super Mario Bros. - IGN." *IGN*, 14 Sept.
 2010, www.ign.com/articles/2010/09/14/ign-presents-the-history-of-super-mario-bros.

Media College. "Quantization." *Mediacollege.com*, 2020,
www.mediacollege.com/glossary/q/quantization.html.

Merriam Webster. "Definition of SINE WAVE." *Merriam-Webster.com*, 2019,
www.merriam-webster.com/dictionary/sine%20wave.

Music Tech Student. "Music Production | Glossary | A.D.S.R." *Musictechstudent.co.uk*, 22
 May 2017, musictechstudent.co.uk/music-production-glossary/adsr/.

Musich Tech Student. "History and Development of Compression - Music Tech Student."

Musictechstudent.co.uk, 11 Aug. 2013,

musictechstudent.co.uk/music_technology_/history-and-development-of-compression

/. Accessed 31 Mar. 2022.

O'Bannon, Richard. "The Power of Musical Repetition." *Wwww.bsomusic.org*, 2022,

[www.bsomusic.org/stories/the-power-of-musical-repetition/#:~:text=%E2%80%9CRe](http://www.bsomusic.org/stories/the-power-of-musical-repetition/#:~:text=%E2%80%9CRepetition%20changes%20the%20way%20we)

[petition%20changes%20the%20way%20we](http://www.bsomusic.org/stories/the-power-of-musical-repetition/#:~:text=%E2%80%9CRepetition%20changes%20the%20way%20we).

Physics Lab. "PhysicsLAB: Introduction to Sound." *Wwww.physicslab.org*, 2022,

[www.physicslab.org/Document.aspx?doctype=3&filename=WavesSound_IntroSound](http://www.physicslab.org/Document.aspx?doctype=3&filename=WavesSound_IntroSound.xml#:~:text=In%20general%2C%20low%20frequency%20waves)

[.xml#:~:text=In%20general%2C%20low%20frequency%20waves](http://www.physicslab.org/Document.aspx?doctype=3&filename=WavesSound_IntroSound.xml#:~:text=In%20general%2C%20low%20frequency%20waves).

Plugin Boutique. "Random Generators Bundle." *Wwww.pluginboutique.com*, 2022,

www.pluginboutique.com/product/2-Effects/39-FX-Bundle/8360-Random-Generators

[-Bundle](http://www.pluginboutique.com/product/2-Effects/39-FX-Bundle/8360-Random-Generators). Accessed 1 Apr. 2022.

Polyphonic. "A Brief History of Electric Guitar Distortion." *Wwww.youtube.com*, 20 Sept.

2018, www.youtube.com/watch?v=iYU90XajYmU&ab_channel=Polyphonic.

Accessed 31 Mar. 2022.

Reid, Gordon. "An Introduction to Additive Synthesis." *Wwww.soundonsound.com*, June 2000,

www.soundonsound.com/techniques/introduction-additive-synthesis.

Robinson, Andrew. "Max MSP -- the Absolute Basics (Tutorial Video Pt 1)."

Wwww.youtube.com, 21 Jan. 2021,

www.youtube.com/watch?v=S7d9TICKXrY&ab_channel=AndrewRobinson.

Rothstein, Joseph. *MIDI : A Comprehensive Introduction*. Madison, Wis., A-R Editions,

1992.

Stanford University. "Schroeder Reverberators." *Stanford.edu*, 2010,

ccrma.stanford.edu/~jos/pasp/Schroeder_Reverberators.html.

Stolet, Jeffrey. "Electronic Music Interactive, 2nd Edition." *Uoregon.edu*, 2019,
pages.uoregon.edu/emi/31.php.

Stolet, Jeffrey. "Electronic Music Interactive, 2nd Edition." *Pages.uoregon.edu*, 2022,
pages.uoregon.edu/emi/17.php#:~:text=17.-. Accessed 1 Apr. 2022.

The physics classroom. "The Doppler Effect." *Physicsclassroom.com*, 2019,
www.physicsclassroom.com/class/waves/Lesson-3/The-Doppler-Effect.

Thomann. "Thomann Oscillator Modules." *Musikhaus Thomann*,
www.thomann.de/gb/oscillator_modules.html?oa=prd&gk=TASYMOOS&filter=true.
Accessed 1 Apr. 2022.

White, Paul. "Sound Synthesis: Part 1." *Www.soundonsound.com*, Feb. 1994,
www.soundonsound.com/techniques/sound-synthesis-part-1#:~:text=Noise%20is%20useful%20for%20simulating. Accessed 1 Apr. 2022.

Yamaha Corporation. "DX7 - Yamaha Design - Yamaha Corporation." *Www.yamaha.com*,
www.yamaha.com/en/about/design/synapses/id_009#:~:text=Released%20in%201983%2C%20this%20digital.