Casio VZ Virtual Instrument: A Replica of the Casio VZ-1/VZ-10M Music Synthesizer

Matthias Wolff [0000–0002–3895–7313]

BTU Cottbus-Senftenberg

May 31, 2020

See https://github.com/matthias-wolff/Casio-VZ-virtual-instrument/blob/master/Casio-VZ-virtual-instrument.pdf for the latest version of this document.

Abstract

In this project I try to rebuild the vintage Casio VZ-1/VZ-10M music synthesizer in Reaktor 6 [6]. The primary goal is a fully functional player which is compatible with MIDI editor/librarian software like Midi Quest [8] or the like. My workplan is

1. make some debugging and development tools (waveform validator, envelope validator, etc.),
2. reproduce the 8 core waveforms of VZ-1/VZ-10M (1x sine, 5x sawtooth-like waveforms created by Casio's Phase Distortion Modulation, 1x white noise, 1x pitch-sensitive narrow-band noise),
3. implement the core sound engine (wavetable oscillators, phase and ring modulators, VCAs, oscillator circuits),
4. implement control signal generators (amplitude envelope, key following, layering, parametric sensitivity characterisitcs, etc.),
5. implement MIDI SysEx control capability, and
6. reproduce the factory voice and operation libraries.

I always strongly disliked the unpleasant—though most characteristic—aliasing and analog noise of the VZ-1. Hence, I will not attempt to reproduce this. Insofar, the remake is not intended to be perfect.
As a secondary goal I may want to reproduce the GUI of the original instrument. This would be a nice-to-have, however not necessarily of much practical use.

Contents

1 Goals and Prerequisites

In this project I try to rebuild the vintage Casio VZ-1/VZ-10M music synthesizer in Reaktor 6 [6]. The primary goal is a fully functional player which is compatible with MIDI editor/librarian software like Midi Quest [8] or the like. My workplan is

1. make some debugging and development tools (waveform validator, envelope validator, etc.),
2. reproduce the 8 core waveforms of VZ-1/VZ-10M (1x sine, 5x sawtooth-like waveforms created by Casio's Phase Distortion Modulation, 1x white noise, 1x pitch-sensitive narrow-band noise),
3. implement the core sound engine (wavetable oscillators, phase and ring modulators, VCAs, oscillator circuits),
4. implement control signal generators (amplitude envelope, key following, layering, parametric sensitivity characterisitcs, etc.),
5. implement MIDI SysEx control capability, and
6. reproduce the factory voice and operation libraries.

I always strongly disliked the unpleasant—though most characteristic—aliasing and analog noise of the VZ-1. Hence, I will not attempt to reproduce this. Insofar, the remake is not intended to be perfect.

As a secondary goal I may want to reproduce the GUI of the original instrument. This would be a nice-to-have, however not necessarily of much practical use.

1.1 The VZ-1/VZ-10M Music Synthesizer

[**TODO:** ...]

1.2 The Reaktor 6 Modular DSP Lab

[**TODO:** ...] [6]

1.3 The Midi Quest 12 Editor/Librarian Software

[**TODO:** ...] [8]

2 Development and Debugging Tools

2.1 Waveform Validator

[**TODO:** ...]

2.2 Envelope Validator

[**TODO:** ...]

3 Development and Debugging Tools

4 Core Sound Engine

VZ's sound engine is called "iPD (interactive phase distortion) modular sound engine" [2, p. 12]. According to Casio, this is an enhanced version of the phase distortion modulation introduced by the predecessor of the VZ-1 called CZ-1 [3, p. 39]. Actually, the VZ synths series seem to feature wavetable oscillators—with waveforms partly generated by the original PD synthesis—together with ordinary phase modulation and ring modulation.

4.1 Theory

I assume that the reader is familiar with the basics of signal processing. Introductions can be found in [5] and [7].

4.1.1 Tones, Timbre, Frequency, and Pitch

A musical tone is a periodic osciallation of sound pressure. Mathematically, we express musical tones as real functions

$$x : \mathbb{R} \to \mathbb{R}$$

which assign signal values $x(t) \in \mathbb{R}$ to instants of time $t \in \mathbb{R}$ measured in seconds. We will call such functions "signals". As a musical tone is periodic, the corresponding signal must be periodic as well. Mathematically, we express this by the condition

$$x(t + nT) = x(t) \qquad \text{with } n \in \mathbb{Z}, \tag{1}$$

where T, measured in seconds, is called the cycle duration. Eq. (1) simply means, that the signal values repeate after an integer number of cylces ist completed. The reciprocal of the cycle duration is called tone or fundamental frequency

$$f = \frac{1}{T}.$$

Frequencies are measured in Hertz (Hz).

The most basic musical tone is a pure tone. Its mathematical representation as a signal is

$$x_{\cos}(t) = a_1 \cos(2\pi f t + \varphi_1), \tag{2}$$

where a_1 denotes the amplitude which is connected to the perceived loudness, f denotes the tone frequency which is connected to the perceived pitch, and φ_1 denotes a phase offset which does not directly relate to perception. Fig. 1a shows the plot of a cosine signal. As the cosine function is periodic, $\cos(\alpha + 2\pi n) = \cos(\alpha)$, so is the cosine signal. The cycle duration of the cosine signal is $T = 1/f$. Note, that the mathematical concept of signals abstracts from sound pressure oscillations. We can also use it for oscillations of other physical quantities like voltages and even for any digital representations of such quantities.

As $\cos(\alpha) = \sin(\alpha + \pi/2)$, pure tones can also be expressed by the sine function

$$x_{\cos}(t) = a_1 \sin\Big(2\pi f t + \varphi_1 + \frac{\pi}{2}\Big).$$

Cosine and sine tones are indistinguishable to the human ear. Hence, both are commonly referred to as "sine" tones.

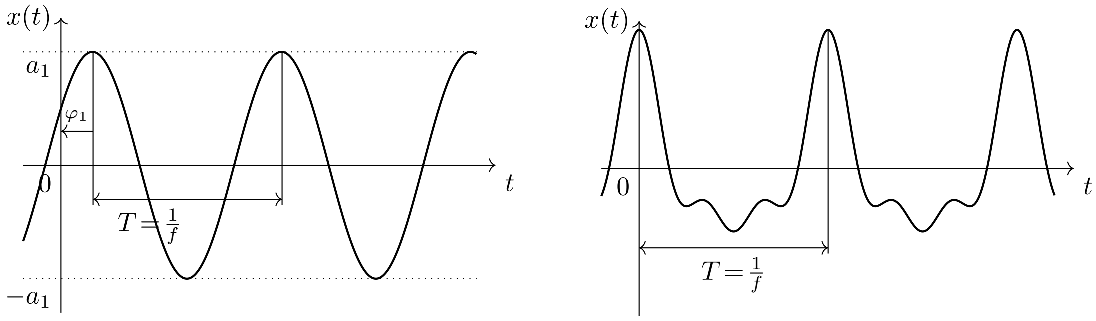
General musical tones are mixtures of pure tones. We express them mathematically as a sum of cosine tones

$$x(t) = \sum_{n=1}^{\infty} a_n \cos\big(2\pi n f t + \varphi_n\big) \tag{3}$$

The summand with $n = 1$ describes the fundamental tone, the summands with $n > 1$ describe so called overtones or partials. The frequencies of all overtones are integer multiples $n \cdot f$ of the tone frequency f. General muscial tones according to Eq. (3) are still periodic with a cycle duration of $T = 1/f$. The fundamental tone and all overtones are characterized by individual amplitudes a_n and phase offsets φ_n. The amplitude-phase tuples (a_n, φ_n) are called (real) FOURIER coefficients and Eq. (3) is called FOURIER series (see, e.g., [5, p. 102–113]).[1] The perceived timbre of a musical tone is mainly connected to the sequence (a_n) of amplitudes. The sequence of phase offsets (φ_n) is perceptually much less relevant than the ampltitudes. Fig. 1b shows an example of a muscial tone composed by the mixture of three pure tones.

Even more complex tones comprise aperiodic components (noise) and varying timbre, i.e. FOURIER coefficients, over time.

[1] Actually, the real FOURIER series contains a further summand, a_0, allowing for a DC offset which we deliberately exclude here as such offsets are undesirable for musical tones.



(a) A cosine tone
$x_{\cos}(t) = a_1 \cos(2\pi f t + \varphi_1)$

(b) A musical tone composed of three cosines
$x(t) = \frac{2}{3} \sum_{n=1}^{2} \frac{1}{n} \cos(2\pi n f t)$

Figure 1: Examples for basic musical tones. The tone frequency—measured in Hertz—is denoted by f. We denote time by t and the duration of one cycle by T. Both are measured in seconds. The argument of the cosine functions is called phase angle φ. One cycle corresponds to a phase difference of $\Delta\varphi = 2\pi$.

Human perception of sound frequency is nearly logarithmic. Therefore, Western music theory commonly uses the pitch

$$p = 69 + 12 \log_2\left(\frac{f}{440\,\mathrm{Hz}}\right) \qquad \texttt{p=vVZtools.f2p(f)} \tag{4}$$

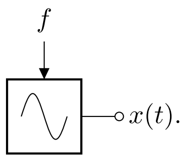
measured in semitones rather than frequencies. The value of 440 Hz in the denominator is the concert pitch. The offset of 69 makes the pitch computed by Eq. (4) compatible with a MIDI frequency data value. According to the MIDI tuning standard [**TODO:** []], p takes values from the interval $0.00 \leq p \leq 127.99$ with the integer part corresponding to a note number and the decimal fraction, with two digits precision, to a detune. MIDI note number $p = 0$ stands for C_{-1}, $p = 69$ for the concert pitch A_4, and $p = 127$ stands for B_9.

Solving Eq. (4) for f, we obtain a formula for converting pitch to frequency:

$$f = 2^{\frac{p-69}{12}} \cdot 440\,\mathrm{Hz} \qquad \texttt{f=vVZtools.p2f(p)}. \tag{5}$$

4.1.2 Oscillators and Waveforms

The basic building blocks of music synthesizers are oscillators. An oscillator is an apparatus, hardware or software, which produces a signal $x(t)$ depending on a single frequency f. As circuit symbol for an oscillator we will use



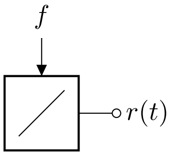
In larger circuit diagrams we will omit the frequency input at the top for legibility.

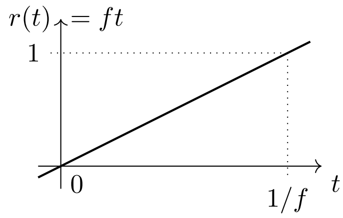
In software synthesizers, the most basic oscillator type is a so called ramp oscillator. A ramp oscillator generates the ramp signal

$$r(t) = ft,$$

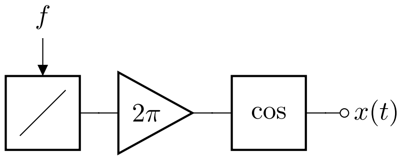
where t denotes time and f a frequency.[2] The circuit symbol and output signal look as follows

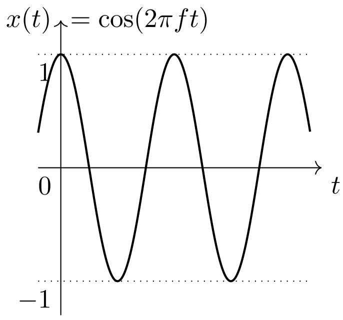
[2]The term "oscillator" is somewhat misleading here as a ramp signal does not oscillate. Yet, that name is commonly used.



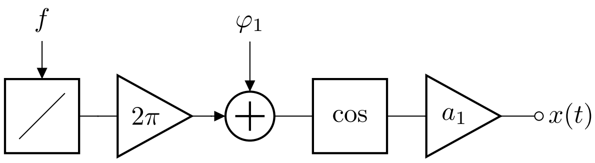


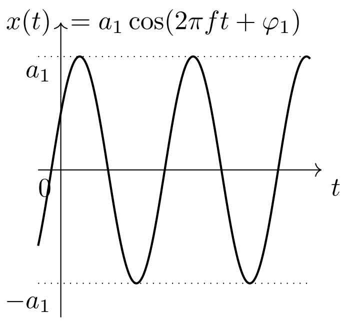
Amplifying the output of a ramp oscillator by a gain factor of 2π and feeding the result into a cosine function, we obtain an oscillator circuit for a pure cosine tone with amplitude 1 and without phase offset:



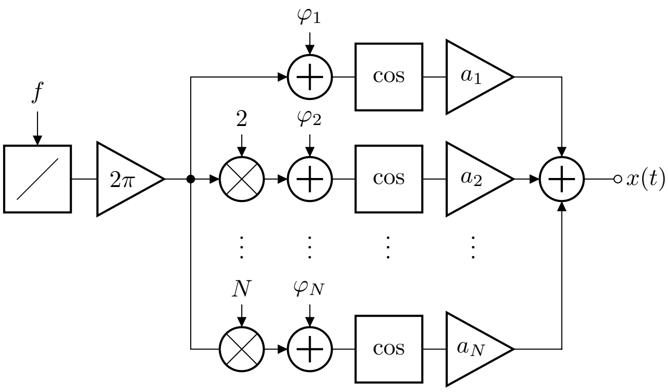


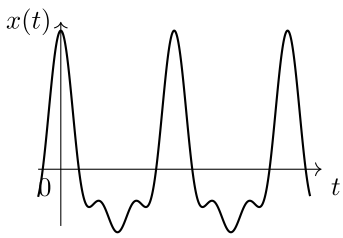
Adding anothor amplifier at the output and a phase offset at the input of the cosine function yields an oscillator circuit for pure tone according to Eq. (2):





In order to add overtones according to Eq. (3), we simply mix the outputs of N cosine oscillators running at frequencies $n \cdot f$ with $1 \leq n \leq N$ as follows:





Such an osciallator circuit is called a “sinusoidal oscillator”. Its sound synthesis paradigm is, for abvious reasons, called “additive”. Sinusoidal oscillators are a direct realization of the Fourier serial according to Eq. (3) truncated to N terms. However, the circuit is quite intricate. Therefore, we simplify things by introducing the concept of waveforms. A waveform is a function

$$\mathrm{w} : \mathbb{R} \to [-1, 1], \quad x = \mathrm{w}(\alpha)$$

Name	Waveform	FOURIER coefficients
Sine	[**TODO:** ...]	
Rectangle	[**TODO:** ...]	
Triangle	[**TODO:** ...]	
Sawtooth	[**TODO:** ...]	

Table 1: Examples of typical waveforms used in music synthesizers

which takes an angle α as argument and which satisfies the following conditions:

$$\mathrm{w}(\alpha + 2\pi n) = \mathrm{w}(\alpha) \qquad \text{w is periodic with a cycle length of } 2\pi, \tag{6}$$

$$-1 \leq \mathrm{w}(\alpha) \leq 1 \qquad \text{w takes values from the interval } [-1, 1], \tag{7}$$

$$\max_{\alpha \in [0, 2\pi]} \big(|\,\mathrm{w}(\alpha)|\big) = 1 \qquad \text{the amplitude of w is 1, and} \tag{8}$$

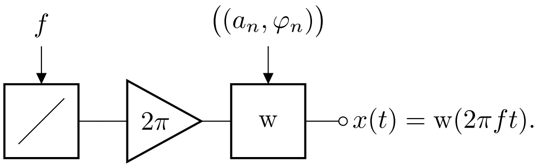
$$\int_0^{2\pi} \mathrm{w}(\alpha)\,\mathrm{d}\alpha = 0 \qquad \text{w has no DC offset.} \tag{9}$$

Clearly, the sine and cosine functions are waveforms by this definition. A general formula for arbitrary waveforms can be derived from the definition of musical tones according to Eq. (3). We set $\alpha = 2\pi f t$ and obtain

$$\mathrm{w}(\alpha) = \sum_{n=1}^{\infty} a_n \cos(n\alpha + \varphi_n). \tag{10}$$

The periodicity and DC conditions (6, 9) are implied by Eq. (10). In order to fulfill the amplitude conditions (7, 8), the FOURIER coefficients a_n and φ_n must meet certain criteria which we will not discuss in detail. Table 1 shows some examples of typical waveforms used in music synthesizers.

The circuit diagram of waveform oscillator producing arbitrary musical tones is

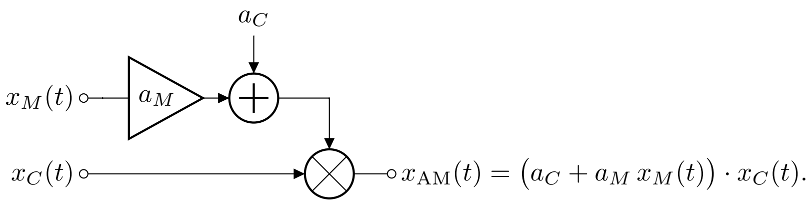


As explained in Section 4.1.1, the timbre of the generated tone can be adjusted by the choice of FOURIER coefficients $\big((a_n, \varphi_n)\big)$.

[**TODO:** excursus to samplers?]

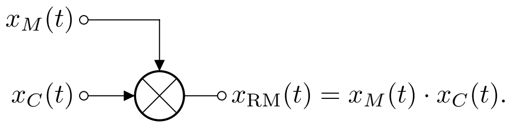
4.1.3 Amplitude and Ring Modulation

By the term "modulation" we denote techniques of manipulating properties of a so called carrier signal $x_C(t)$ by a modulating signal $x_M(t)$. Amplitude modulation refers to modifiying the carrier's amplitude as follows:



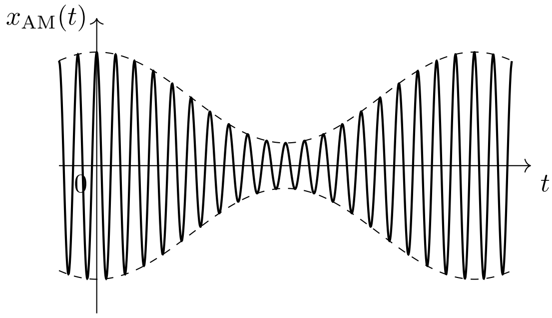
(11)

Ring modulation is a related concept. Here, the gain factor a_C of the carrier is zero and the gain factor a_M of the modulating signal is one which yields

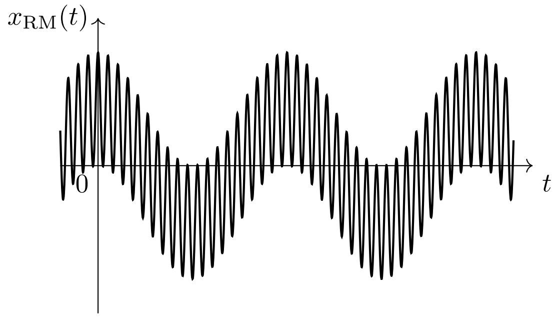


(12)

Fig. 2 shows examples for amplitude and ring modulation of two pure cosine tones.



(a) Amplitude modulation according to Eq. (11) with $a_C = 0.6$, $a_M = 0.4$, $x_C(t) = \cos(2\pi f t)$, and $x_M(t) = \cos\big(2\pi(0.05 f)t\big)$



(b) Ring modulation according to Eq. (12) with $x_C(t) = \cos(2\pi f t)$ and $x_M(t) = \cos\big(2\pi(0.9 f)t\big)$

Figure 2: Examples of amplitude and ring modulation of two cosine waveforms

4.1.4 Phase and Frequency Modulation

For phase and frequency modulation we restrict the carrier signal to be a waveform

$$x_C(t) = \mathrm{w}_C(2\pi f t), \tag{13}$$

where we chose $\alpha = 2\pi f t$ as the angle argument. *Phase modulation* is modifying the phase angle of the carrier waveform by a time-varying phase shift signal $\varphi(t)$

$$x_{\mathrm{PM}}(t) = w_C\big(2\pi f t + \varphi(t)\big).$$

If whe choose another waveform, w_M, with a *different* frequency f_M and amplified by a gain factor $2\pi a_M$ as the phase shift signal

$$\varphi(t) := 2\pi a_M \, w_M(2\pi f_M t),$$

we obtain a phase-modulated signal

$$x_{\mathrm{PM}}(t) = w_C\big(2\pi f t + 2\pi a_M \, w_M(2\pi f_M t)\big). \tag{14}$$

For the following considerations we note, that the angle argument $2\pi f t$ of the carrier waveform (13) is a time-varying or, in other words, a phase *signal*

$$\varphi(t) = 2\pi f t. \tag{15}$$

Actually, it is a linear function of time t with a frequency-dependent slope $2\pi f$ an zero offset. Hence we can write the carrier waveform as

$$x_C(t) = w_C\big(\varphi(t)\big) \quad \text{with } \varphi(t) = 2\pi f t. \tag{16}$$

By definition, frequency is proportional to the time derivative of phase

$$f(t) = \frac{1}{2\pi} \frac{\mathrm{d}\,\varphi(t)}{\mathrm{d}\,t}. \tag{17}$$

In general, also frequency may be time-varying, i.e., a frequency *signal.* The values of frequency signals according to Eq. (17) are called instantaneuos frequencies. Resolving Eq. (17) for $\varphi(t)$ yields

$$\varphi(t) = 2\pi \int\limits_0^t f(\tau)\,\tau + \varphi_0, \tag{18}$$

where τ denotes an auxiliary time variable—which is only needed because t is the upper limit of the integral—and φ_0 denotes an integration constant which we will ignore in the following.

The concept of instantaneous frequencies generalizes the fixed frequencies we used so far for our waveforms. We can see this easily by applying Eq. (17) to Eq. (15):

$$f(t) = \frac{1}{2\pi} \frac{\mathrm{d}\,\varphi_C(t)}{\mathrm{d}\,t} = \frac{1}{2\pi} \frac{\mathrm{d}\,(2\pi f t)}{\mathrm{d}\,t} = f.$$

Frequency modulation is characterized by a time-varying frequency of the carrier waveform

$$f(t) = f + f_M(t) \tag{19}$$

which is the sum of the constant tone frequency f and a time-varying frequency shift $f_M(t)$. Again, we choose another waveform, w_M, with a *different* frequency f_M and amplified by a gain factor $2\pi a_M$ as the modulating signal

$$f_M(t) := a_M\, w_M(2\pi f_M t)$$

and obtain

$$f(t) = f + a_M\, w_M(2\pi f_M t) \tag{20}$$

as instantaneous frequency of our carrier signal. Integrating Eq. (20) yields the respective phase signal

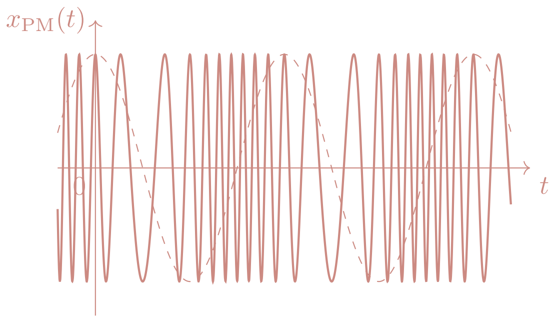
$$\varphi(t) = 2\pi \int_0^t f(\tau)\,\mathrm{d}\tau = 2\pi f t + 2\pi\, a_M \int_0^t w_M(2\pi f_M \tau)\,\mathrm{d}\tau \tag{21}$$

which we can introduce into Eq. (16) to obtain a formula for a frequency-modulated waveform:

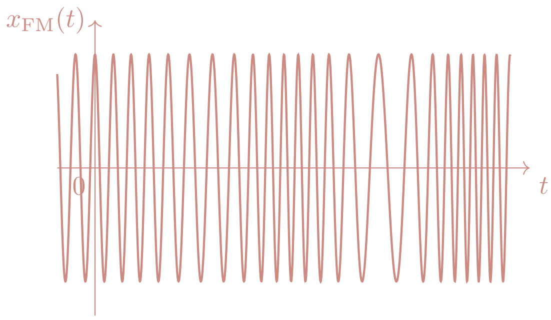
$$x_{FM}(t) = w_C\left(2\pi f t + 2\pi\, a_M \int_0^t w_M(2\pi f_M \tau)\,\mathrm{d}\tau\right). \tag{22}$$

Let's finally compare Eqs. (14) and (22). We see that phase and frequency modulation are closely related. Frequency modulation is actually just a phase modulation with the integral over the modulating signal. For this reason, many "FM" synthesizers actually perform a phase modulation, as the latter is much easier to implement and "real" FM introduces—by integrating over the modulating signal—undesirable changes of the timbre with note frequency [**TODO:** []].

Fig. 3 shows simple examples of phase and frequency modulation.



(a) Phase modulation according to Eq. (14) with $w_C(t) = \cos(2\pi f t)$, $w_M(t) = \cos\big(2\pi(0.1 f)t\big)$, and $a_M = 0.1$



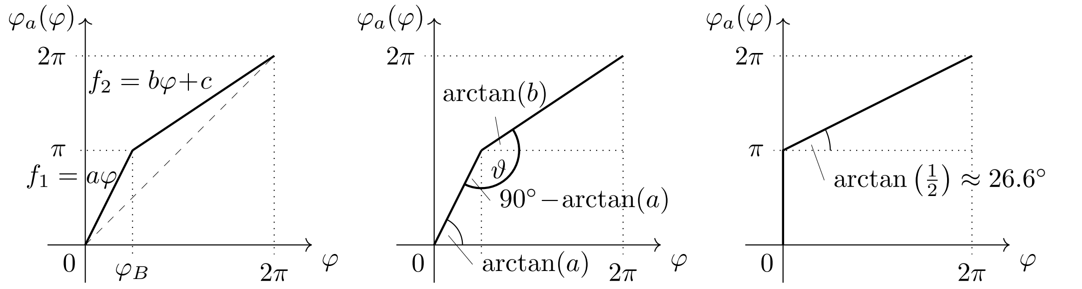
(b) Frequency modulation according to Eq. (22) with $a_C = 0.6$, $a_M = 0.4$, $x_C(t) = \cos(2\pi f t)$, and $x_M(t) = \cos\big(2\pi\,(0.05 f)\,t\big)$, i.e., $f_M = 0.05\, f_C$

Figure 3: Examples of phase and frequency modulation of two cosine waveforms

4.1.5 Phase Distortion Modulation

Phase distortion modulation (PDM) was invented in the early 1980's by I. Tomita, Y. Takahashi and collegues of Casio Computer Ltd. for the CZ music synthesizer series [4]. Fig. 6 shows an example from the CZ manual. The VZ series uses waveform oscillators based on PDM.

Core element of PDM is a phase distortion characteristic curve as shown in Fig. 4a [1, p. 20–21]. We see that the curve consists of two straight sections



(a) Piecewise linear phase charateristics

(b) Distortion angle ϑ, $135° \leq \vartheta \leq 180°$

(c) Minimum distortion angle $\vartheta_{\min} \approx 116.6\,°$, $a \to \infty$

Figure 4: Characteristic curve of phase distortortion

$$f_1(\varphi) = a\varphi, \qquad 0 \leq \varphi < \varphi_B, \quad \text{connecting points } (0,0) \text{ and } (\varphi_B, \pi), \text{ and} \tag{23}$$

$$f_2(\varphi) = b\varphi + c, \quad \varphi_B \leq \varphi < 2\pi, \quad \text{connecting points } (\varphi_B, \pi) \text{ and } (2\pi, 2\pi). \tag{24}$$

The slope a of the first section is variable and can take real values ≥ 1. The start and end points of the sections define the following constraints

$$f_1(\varphi_B) = a\varphi_B = \pi \qquad \rightsquigarrow \varphi_B = \frac{\pi}{a} \tag{25}$$

$$f_2(\varphi_B) = \frac{\pi}{a}b + c = \pi \tag{26}$$

$$f_2(2\pi) = 2\pi b + c = 2\pi \tag{27}$$

From Eqs. (25) to (27) follows

$$b = \frac{a}{2a-1} \quad \text{and} \quad c = \frac{2a-2}{2a-1}\pi, \tag{28}$$

and finally

$$\varphi_a(\varphi) = \begin{cases} a\varphi & \text{for } 0 \leq \varphi < \frac{\pi}{a} \\ \dfrac{a\varphi + 2\pi(a-1)}{2a-1} & \text{for } \frac{\pi}{a} \leq \varphi < 2\pi \end{cases} \quad \text{with } a \geq 1 \qquad \texttt{phipd=vVZtools.PDMcc(phi,a).} \tag{29}$$

Remark: An alternate phase distortion characteristic $f_a(x)$ with domain $0 \leq x < 1$ taking values $0 \leq f_a(x) < 1$ may be needed for Reaktor implementation:

$$f_a(x) = \begin{cases} ax & 0 \leq x < \frac{1}{2a} \\ \dfrac{ax + a - 1}{2a-1} & \frac{1}{2a} \leq x < 1 \end{cases} \qquad \text{with } a \geq 1. \tag{30}$$

Modulation depth can also be expressed by the angle between the straight sections

$$\begin{aligned} \vartheta &= \pi - \arctan(a) + \arctan(b) \\ &= \pi - \arctan(a) + \arctan\left(\frac{a}{2a-1}\right) \qquad \texttt{theta=vVZtools.pdmA2theta(a)} \end{aligned} \tag{31}$$

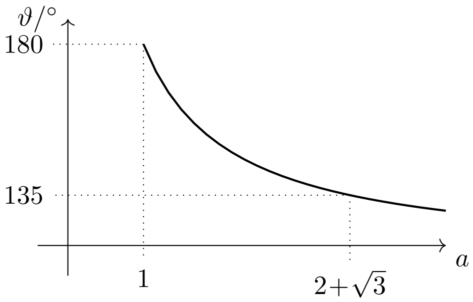
measured in degrees (see Fig. 4b). Literature states that the permissible range is $135° \leq \vartheta \leq 180°$ [4]. However, for the wavetable of the VZ series this restriction does not seem to hold. Measurement yielded angles down to 123° corresponding to slope parameters $a > 10$.[3] As I did not manage to solve Eq. (31) for a, I just plot the function and print a lookup table in Fig. 5.

Waveforms `SINE` and `SAW1`–`SAW5` of VZ-1/VZ-10M resemble phase-distorted cosine waves

$$w(\varphi) = \cos\big(\varphi_a(\varphi)\big). \tag{32}$$

Figure 6 shows some examples.

[3]see `matlab/PhaseDistortionModulationAnalysis.mlx`

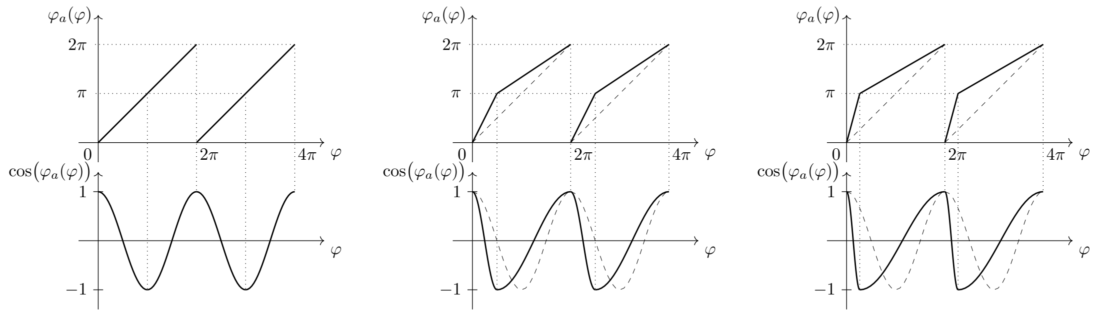


(a) Distortion angle parameter ϑ as a function of slope parameter a

$\vartheta/°$	a	$\vartheta/°$	a
116.6	∞	150	2.02
120	20.07	155	1.74
125	8.19	160	1.52
130	5.14	165	1.35
135	$2+\sqrt{3} \approx 3.73$	170	1.21
140	2.92	175	1.10
145	2.39	180	1.00

(b) Lookup table

Figure 5: Relation between distortion angle parameter ϑ and slope parameter a, see Eq. (31)



(a) $\vartheta = 180°$ $(a = 1)$ no modulation

(b) $\vartheta = 150°$ $(a \approx 2)$

(c) $\vartheta = 135°$ $(a \approx 3.73)$ resembling a sawtooth signal

Figure 6: Example of phase distortion modulation of a cosine adapted from [1, p. 20–21, Figs. 1–3]. Phase angles $\varphi < 0$ and $\varphi \geq 2\pi$ are wrapped into the base interval $0 \leq \varphi < 2\pi$. Note that [1] uses $-\cos(\varphi)$ as carrier signal.

4.1.6 Wavetable Osciallators

The original VZ-1/VZ-10M is apparently based on wavetable oscillators [2, p. 34]. Such oscillators contain a set of digital waveforms, each comprising exactly one signal period. Depending on the pitch, these waveforms are played back at different rates. **[TODO: ??]** shows an example of a VZ waveform (`SAW4`).

Formally, a single waveform can be expressed by

$$w(\varphi) \quad \text{with } 0 \leq \varphi < 2\pi, \tag{33}$$

where the argument φ is a phase angle. A oscillator based on this waveform uses a time-varying phase angle $\varphi(t)$. Its output signal can be written as

$$x(t) = w\big(\varphi(t)\big) = w\big(2\pi(f_0 t \mod 1)\big) \quad \text{with } -1 \leq x(t) \leq 1 \tag{34}$$

where w denotes a waveform, $f_0 \in \mathbb{R}^{>0}$ denotes the note frequency in Hertz, and t denotes the time in seconds. Further, we denote by

$$x \mod 1 := x - \lfloor x \rfloor \quad \text{with } 0 \leq (x \mod 1) < 1 \tag{35}$$

the decimal fraction of the real valued quantity x. Hence, the argument $\varphi(t) = 2\pi(f_0 t \bmod 1)$ on the right side of Eq. (34) involves resetting the phase angle of the waveform to zero whenever the cycle length 2π is surpassed. **[TODO: ??]** shows an example.

We will use the circuit symbol **[TODO: ...]** for a wavetable oscillator.

4.2 VZ Hardware

4.2.1 Waveforms

The VZ series features eight waveforms: sine (`SINE`), five sawtooth-shaped waveforms (`SAW1`–`SAW5`), white noise (`NOISE1`), and a mixture of white noise and a sine tone (`NOISE2`). The `SAWn` waveforms resemble phase-distortion modulated cosines

$$x(\varphi) = \cos\left(\varphi_a(\varphi)\right) = \sin\left(\varphi_a(\varphi) + \frac{\pi}{2}\right), \tag{36}$$

where $\varphi_a(\varphi)$ denotes the phase distortion characteristic according to Eq. (29) with slope parameter a. Thorough analysis of recordings (see `matlab/PhaseDistortionModulationAnalysis.mlx`) shows that VZ's actual `SAWn` waveforms exhibit phase shifts of the carrier signal and—even worse—the measured phase distortion characteristics is *not* pieceweise linear. [**TODO: ??**] shows an example. I conclude that VZ-1 does *not* employ PDM as described in [1] but some approximation of it. Consequently, I have two options for the replica

- purist: implement a PDM oscillator anyway or
- thruthful: implement a wave table oscillator using wave cycles sampled from the VZ-1.

4.3 Replica

4.3.1 Voltage-Controlled Oscillators (VCO)

In order to read the wavetable out of my VZ-1, I proceeded as follows

1. Set the master tune of VZ-1 to 440 Hz in `MENU 3-00`,
2. Initialize a new voice,
3. Mute all oscillator modules except `M1` and configure line `A` as `MIX` in `MENU 1/00`,
4. In `MENU 1/02`, set `M1` to a fix pitch of `OCT=00`, `NOTE=00`, `FINE=08` to generate tones with a frequency of $46^7/_8$ Hz (see below),
5. Configure all amplitude controls of `M1` for maximum signal output and set a trivial amplitude envelope mimicking a mere gate signal,
6. Disable all LFOs and MIDI controllers on `M1`.

I recorded with the Komplete Audio 6 sound interface at a sampling frequency of $f_S = 48\,\text{kHz}$. Each waveform in the wavetable shall contain $K = 1024$ samples for one full cycle. Hence, the "natural" frequency of the waveforms—i.e., when reading out at 48 kSamples per second—is

$$f_0 = \frac{f_A}{K} = \frac{48000\,\text{Hz}}{1024} = 46^7/_8\,\text{Hz} \tag{37}$$

which corresponds to the musical note F#1 + 23 ct. Using the fix pitch settings above, one cycle of the waveform being recorded is exactly 1024 samples long and no resampling is necessary.

4.3.2 Voltage-Controlled Amplifiers (VCA)

5 Control Signal Generators

6 MIDI SysEx Control

Issue: Reaktor 6 does not seem to support MIDI SysEx. Circumvent by control automation?

7 GUI Replica

- LCD dot matrix display 96×64 pixels

References

[1] Casio Computer Co., Ltd. *Casio CZ-1 Operation Manual*, 1987. Available online: https://www.manualslib.com/download/1156999/Casio-Cz-1.html.

[2] Casio Computer Co., Ltd. *Casio VZ-1 User Manual*, 1988. Available online: https://www.manualslib.com/manual/358586/Casio-Vz-1.html#product-VZ-1.

[3] S. de Furia and J. Scacciaferro. *Power Play VZ - The Essential Guide to Practical Applications*. Hal Leonard Publishing Corp., 1989. ISBN 0-88188-871-0. Available online: https://www.manualslib.com/manual/358586/Casio-Vz-1.html#product-VZ-1.

[4] H. Gerdes. Workshop: Casio CZ/VZ und die Grundlagen der Phase Distortion Synthesis. Online: https://www.amazona.de/workshop-casio-czvz-und-die-grundlagen-der-phase-distortion-synthesis/, Dec. 2009.

[5] Rüdiger Hoffmann and Matthias Wolff. *Intelligente Signalverarbeitung 1 - Signalanalyse*. Springer Vieweg, 2nd edition, 2014. ISBN: 978-3-662-45322-3.

[6] Native Instruments GmbH. Reaktor 6 Modular DSP Lab. Online: https://www.native-instruments.com/en/products/komplete/synths/reaktor-6. retrieved: Aug. 30, 2019.

[7] A. V. Oppenheim and R. W. Schafer. *Discrete-Time Signal Processing*. Pearson, 3rd edition, 2014. ISBN: 978-9332535039.

[8] Sound Quest Inc. Midi Quest 12. Online: https://squest.com/Products/MidiQuest12/index.html. retrieved: Aug. 30, 2019.