ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/267629988

More Cowbell: a Physically-Informed, Circuit-Bendable, Digital Model of the TR-808 Cowbell

Conference Paper · October 2014

CITATION
1

READS
830

3 authors, including:

Kurt James Werner
Queen's University Belfast
35 PUBLICATIONS **120** CITATIONS
SEE PROFILE

Julius Smith
Stanford University
346 PUBLICATIONS **6,471** CITATIONS
SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Physical model of a drum: FDTD techniques combined with a room acoustics model View project

Faust spectral tilt filters View project

All content following this page was uploaded by Kurt James Werner on 01 November 2014.

The user has requested enhancement of the downloaded file.

Audio Engineering Society

Convention Paper 9207

Presented at the 137th Convention
2014 October 9–12 Los Angeles, USA

This paper was peer-reviewed as a complete manuscript for presentation at this Convention. Additional papers may be obtained by sending request and remittance to Audio Engineering Society, 60 East 42nd Street, New York, New York 10165-2520, USA; also see www.aes.org. All rights reserved. Reproduction of this paper, or any portion thereof, is not permitted without direct permission from the Journal of the Audio Engineering Society.

More Cowbell: a Physically-Informed, Circuit-Bendable, Digital Model of the TR-808 Cowbell

Kurt James Werner[1], Jonathan S. Abel[1], and Julius O. Smith[1]

[1] *Center for Computer Research in Music and Acoustics (CCRMA), Stanford Univ., Stanford, CA, 94305, USA*

Correspondence should be addressed to Kurt James Werner (kwerner@ccrma.stanford.edu)

ABSTRACT

We present an analysis of the cowbell voice circuit from the Roland TR-808 Rhythm Composer. A digital model based on this analysis accurately emulates the original. Through the use of physical and behavioral models of each sub-circuit, this model supports accurate emulation of circuit-bent extensions to the voice's original behavior (including architecture-level alterations and component substitution). Some of this behavior is very complicated and is inconvenient or impossible to capture accurately through black box modeling or structured sampling. The band pass filter sub-circuit is treated as a case study of how to apply Mason's gain formula to finding the continuous-time transfer function of an analog circuit.

1. INTRODUCTION

Building on previous work by the authors on analyzing and modeling the TR-808 bass drum circuit [2] and cymbal circuit [3], the goals of this research are to analyze the circuitry of the TR-808's cowbell voice circuit and create physically-informed models of each of its sub-circuits. The analysis will also provide a resources for designers of real world modifications to the circuit.

Modeling each sub-circuit individually exposes the underlying logic of the circuit's overall architecture and enables the simulation of circuit modifications based on architecture-level changes. Modeling the behavior of each sub-circuit in terms of the values of its electrical components exposes the details of each sub-circuit's behavior and makes the simulation of circuit modifications based on component substitution simple. This preserves the potential for mods inherent in the original circuit.

Many of the TR-808's voices were tunable (low/mid/high congas/toms) or had fairly powerful tone-shaping capabilities (bass drum, snare drum, cymbal), but user control of the cowbell voice is limited to a simple level control.

The cowbell circuit produces a sound with two tones, nominally at 540 Hz (C_5+55 cents) and 800 Hz

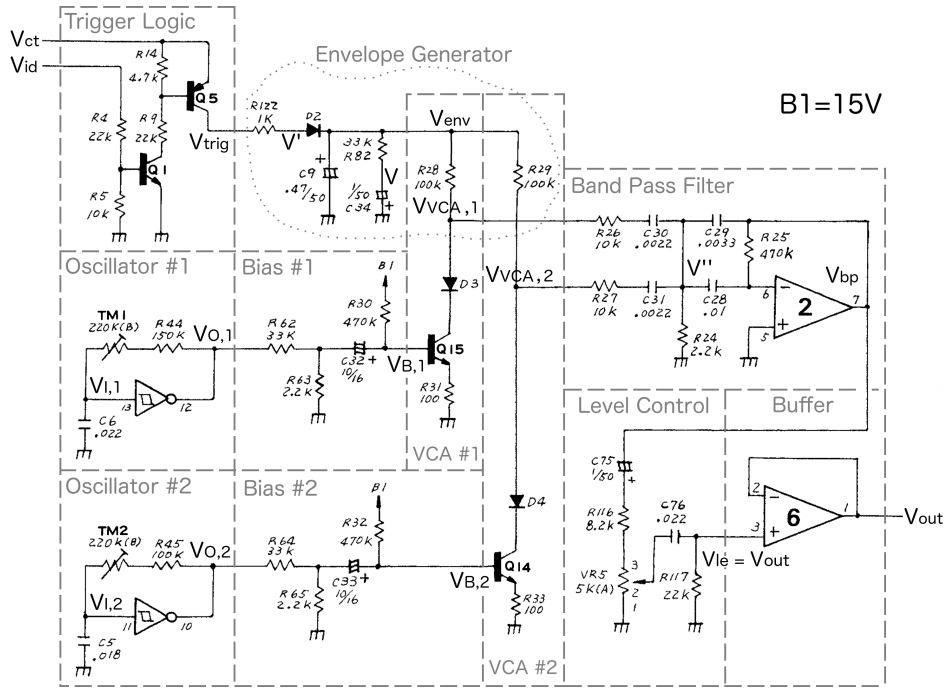

Fig. 1: TR-808 cowbell schematic, blocks marked (adapted from [1]).

(G_5+35 cents). These form an out-of-tune perfect 5th (6.80 semitones). The tuning of the tones is factory-set using two internal trimpots (TM_1 and TM_2). Drum machine hackers were quick to move these out to the front panel of "the 808," expanding the expressive capabilities of the device [4]. These tuning controls are just one example of the latent expressive potential that is built into the cowbell circuit[1], but inaccessible without the introduction of mods.

In the 1980s, the TR-808 cowbell voice was featured in such hit songs as Whitney Houston's "I Wanna Dance With Somebody" and Africa Bambaataa's "Planet Rock." Today, it remains a fixture in R&B, Hip Hop, Techno, and Acid House.

The 808 cowbell's popularity has led to a proliferation of hardware clones. Eric Archer gives a brief explanation of the 808 cowbell's operation and gives a good example of the tradition of cloning and modding the circuit to extend the range of sounds it can create [6]. Gordon Reid discusses the 808 cowbell in the context of imitating its sound with modular analog synthesizers [7].

We give an overview of the circuit in §2 and an analysis of each sub-circuit in §§3–8. This is followed by a discussion of modeling techniques and supported circuit bends and mods in §9. Finally, §10 gives a discussion of results and conclusions.

2. OVERVIEW

Fig. 1 shows a schematic diagram of the TR-808 cowbell circuit [1]. This schematic is annotated with important nodes and component labels, and exposes how the cowbell voice circuit is broken into subcircuits: Schmitt trigger oscillators (see §3), trigger logic (see §4), an envelope generator (see §5), "swing-type voltage-controlled amplifiers" (VCAs, see §6), a band pass filter (see §7), and a level control stage and buffer (see §8).

Fig. 2 shows a block diagram of the digital emulation of the cowbell voice circuit. The bulk of this paper will deal with analyzing the behavior of the sub-

[1]Eric Archer's "TR-808 Cowbell Clone w/ mods" gives a very good example of some of the possible modifications [5].

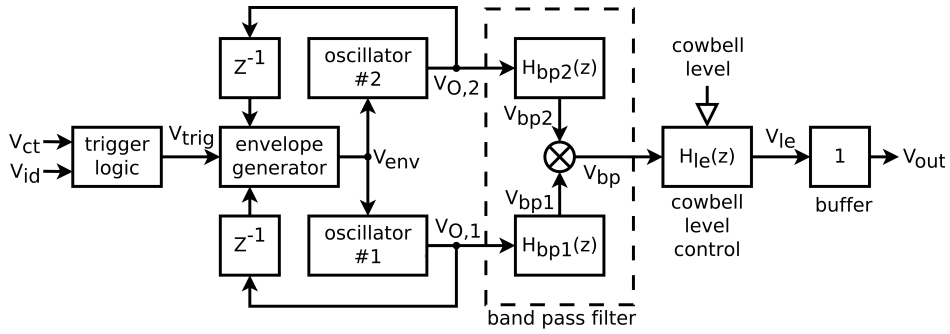

Fig. 2: TR-808 cowbell emulation block diagram.

circuits shown in Fig. 1 to produce a digital model in the form of Fig. 2.

To produce a cowbell note, the CPU applies a common trigger accent level V_{ct} and cowbell instrument timing data V_{id} to the trigger logic. The resulting 1-ms long pulse is delivered to the envelope generator. This envelope is applied to two separate swing-type VCAs, which control the amplitude of two rectangular waves. The output of each VCA is applied to a band pass filter, whose output is applied to a cowbell level control that also acts as an output buffer.

3. SCHMITT TRIGGER OSCILLATORS

The heart of the TR-808 cowbell voice circuit is a pair of rectangular wave oscillators. In each of these astable multivibrators, an inverter with a Schmitt trigger[2] on its input acts as the bistable element, and a passive network of a single resistor and single capacitor provides an RC time constant which tunes the oscillator to a particular frequency.

The electrical components considered are R_{osc} and C_{osc}. The electrical characteristics of the inverter include the low output state V_{OL}, high output state V_{OH}, positive-going threshold voltage V_{T^+}, and negative-going threshold voltage V_{T^-}.[3] For the two rectangular wave oscillators in the cowbell voice circuit, R_{osc} consists of the series combination of a resistor and a trimpot: $R_{osc1} = \mathrm{TM}_1 + R_{44}$ and $R_{osc2} = \mathrm{TM}_2 + R_{45}$.

Assuming an ideal inverter[4], the differential equation relating the input and output voltages of one of the Schmitt trigger oscillators is:

$$\frac{V_O - V_I}{R_{osc}} = V_I C_{osc} s \,. \tag{1}$$

Considering this equation with different sets of boundary conditions and the given switching action of the Schmitt trigger inverter yields capacitor charging and discharging times:

$$t_{\text{charge}} = R_{osc} C_{osc} \cdot \ln\left(\frac{V_{OH} - V_{T^-}}{V_{OH} - V_{T^+}}\right) \tag{2}$$

$$t_{\text{discharge}} = R_{osc} C_{osc} \cdot \ln\left(\frac{V_{OL} - V_{T^+}}{V_{OL} - V_{T^-}}\right). \tag{3}$$

Summing these yields $T = t_{\text{charge}} + t_{\text{discharge}}$, the period of oscillation of the oscillator.

The behavior of each rectangular wave oscillator can be characterized by its frequency, amplitude, and duty cycle. The frequency of each oscillator is the inverse of its period: $f = 1/T$. The amplitude of each oscillator is equal to the difference between its two possible output voltages: $A = V_{OH} - V_{OL}$. Duty

[2]Schmitt triggers (introduced by Otto H. Schmitt in 1937 [8]) are a form of comparator with hysteresis—they have two different switching thresholds, depending on the current state of the trigger. In the case of a Schmitt trigger inverter, the output voltage V_O will swing low to V_{OL} when the input voltage V_I rises above some threshold V_{T^+}, and the output will swing high to V_{OH} when the input falls below some second threshold V_{T^-}.

[3]Assuming a supply voltage of $V_{DD} = 5.0$ V, and a temperature of 25 °C, typical electrical characteristics for the HD14584 Hex Schmitt Inverter chip used in the TR-808 cowbell circuit are $V_{OL} = 0$ V, $V_{OH} = 5$ V, $V_{T^+} = 2.7$ V, and $V_{T^-} = 2.1$ V.

[4]one that can source infinite current and has infinitely fast switching time

cycle is the proportion of time that the oscillator spends in its high state: $D = {}^{t_{charge}}/_{T}$.

Plugging in Equations (2)–(3) to these definitions yields expanded equations for frequency:

$$f = \frac{1}{R_{osc}C_{osc}\left(\ln\left(\frac{V_{OH}-V_{T-}}{V_{OH}-V_{T+}}\right)+\ln\left(\frac{V_{OL}-V_{T+}}{V_{OL}-V_{T-}}\right)\right)} \quad (4)$$

and duty cycle:

$$D = \ln\left(\frac{V_{OH}-V_{T-}}{V_{OH}-V_{T+}}\right)/\ln\left(\frac{(V_{OH}-V_{T-})(V_{OL}-V_{T+})}{(V_{OH}-V_{T+})(V_{OL}-V_{T-})}\right). \quad (5)$$

Note that R_{osc} and C_{osc} drop out of Equation (5)—D depends only on the electrical characteristics of the inverter.

Given the typical electrical characteristics of the HD14584, this yields a duty cycle of $D = 0.4798$. Given the stock component values in the oscillators, this yields a factory tuning range of 254.3–627.2 Hz for oscillator 1 and 359.4–1149.9 Hz for oscillator 2.

4. TRIGGER LOGIC

The trigger logic sub-circuit takes in a timing signal V_{id} and a global accent level V_{ct} (which is applied to all of the voices in the 808) and combines them into a signal V_{trig} which drives the envelope generator.

Although the physics of this circuit are complex, it can be modeled simply. The trigger logic "ANDs" V_{id} and V_{ct}, so that V_{ct} has the timing (1-ms wide) of V_{id} and the amplitude (7–14 V) of V_{ct} [1].

5. ENVELOPE GENERATOR

The cowbell voice circuit's amplitude envelope is the source of much of its character. This behavior comes from the envelope generator, which produces a sharp attack and complex two-stage decay.

By treating the output of the trigger logic and the VCAs (which will be studied in §6) as voltage sources, the behavior of the envelope generator sub-circuit can be described by a system of three first-order, nonlinear differential equations:

$$\frac{V_{trig}-V'}{R_{122}} = I_{ES}\left(e^{\frac{V'-V_{env}}{V_T}}-1\right) \quad (6)$$

$$I_{ES}\left(e^{\frac{V'-V_{env}}{V_T}}-1\right) = V_{env}C_9 s + \frac{V_{env}-V}{R_{82}} + \frac{V_{env}-V_{\mathrm{VCA},1}}{R_{28}} + \frac{V_{env}-V_{\mathrm{VCA},2}}{R_{29}} \quad (7)$$

$$\frac{V_{env}-V}{R_{82}} = VC_{34}s\,. \quad (8)$$

Solving this nonlinear system will involve evaluating the Lambert W function. To avoid the use of computationally-intensive iterative methods, we adopt a simplified diode model that yields a linear (though time-varying) system.

Because the diode in the envelope generator spends the majority of its time away from the corner of its I–V characteristic, we replace it with a simplified model.[5] This way the attack and release phases of the envelope generator can be separated. The equations describing their behavior are simplified and expressed as a system of two ordinary differential equations. During the attack phase (when $V_{trig} \geq V_{env} + V_{on}$), the following equations describe the envelope generator's behavior:

$$\frac{\mathrm{d}V_{env}}{\mathrm{d}t} = V_{\mathrm{VCA},1}\frac{1}{R_{28}C_9} + V_{\mathrm{VCA},2}\frac{1}{R_{29}C_9} + V\frac{1}{R_{82}C_9} - V_{env}\left(\frac{1}{R_{28}C_9}+\frac{1}{R_{29}C_9}+\frac{1}{R_{82}C_9}+\frac{1}{R_{122}C_9}\right) - V_{trig}\frac{1}{R_{122}C_9} + V_{on}\frac{1}{R_{122}C_9} \quad (10)$$

$$\frac{\mathrm{d}V}{\mathrm{d}t} = V_{env}\frac{1}{R_{82}C_{34}} - V\frac{1}{R_{82}C_{34}}\,. \quad (11)$$

During the release phase (when $V_{trig} < V_{env} + V_{on}$), Equation (10) is replaced by:

$$\frac{\mathrm{d}V_{env}}{\mathrm{d}t} = V_{\mathrm{VCA},1}\frac{1}{R_{28}C_9} + V_{\mathrm{VCA},1}\frac{1}{R_{29}C_9} + V\frac{1}{R_{82}C_9} - V_{env}\left(\frac{1}{R_{28}C_9}+\frac{1}{R_{29}C_9}+\frac{1}{R_{82}C_9}\right). \quad (12)$$

In discrete time, this system of equations can be solved numerically using the Forward Euler method.

Fig. 3 shows the effect of various accent levels. Since the family of curves are not exact linear shifts of each other on a log scale, it is clear that the accent level controls the time-evolution of the envelope, as well as its peak amplitude.

6. SWING-TYPE VCAS

Each of the so-called swing-type VCAs perform a sort of non-linear voltage-controlled amplifica-

[5] When the voltage across the diode is less than a standard silicon diode drop, $V_{on} = 0.6$, the diode is treated as an open circuit. When the voltage is greater than a standard diode drop, it is treated as a short:

$$R_{diode} = \begin{cases} \infty & \text{if} \quad V_{diode} < V_{on} \\ 0 & \text{if} \quad V_{diode} \geq V_{on} \end{cases} \quad (9)$$

.

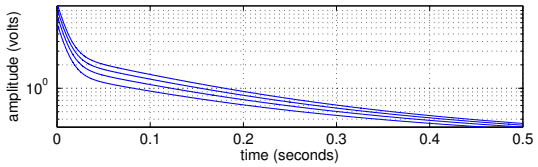

Fig. 3: Family of envelopes V_{env} in response to various accent levels, $V_{ct} = \{8, 10, 12, 14\}\, V$.

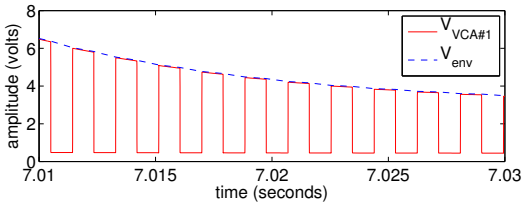

Fig. 4: Swing-type VCA behavior.

tion. They resemble common-emitter amplifiers, albeit ones with non-standard biasing schemes, extra diodes, and an applied envelope voltage replacing V_{CC}. Input signals (the rectangular waves $V_{O,1}$ and $V_{O,2}$ studied in §3) are applied via a biasing sub-circuit, and the envelope generator controls the amount of amplification.

A more complete physically-informed model of the biasing and large-signal behavior of the swing-type VCAs appears in [3], and involves cascading a modified common-emitter amplifier expression into an estimated memoryless, though dynamic (dependent on V_{env}), nonlinearity.[6] The VCAs studied in [3] need to handle arbitrary signal levels and require an accordingly complex model. Leveraging the fact that the input signals applied to the VCAs in the cowbell voice circuit are bi-level signals, we employ a simplified behavioral model that is valid for bi-level signals only. It is both less computationally expensive and more accurate than the model presented in [3]).

6.1. Biasing

Each of the swing-type VCAs has its input (the base of the BJT transistor, Q_{14} or Q_{15}) high pass filtered and biased by a network of resistors and a DC-blocking capacitor.

The behavior of this network is estimated using the method of [9, §3.2]. The frequency response and introduced bias are found by tabulation of a SPICE simulation. Within the audio band (and down to DC), the frequency response is well-approximated by a first order high pass filter with a cutoff frequency $f_c = 5.273$ Hz and a passband attenuation of $G = -28.59$ dB. A positive bias of $V_{bias} = 0.4455$ V develops via R_{28} or R_{29}.[7]

[6] a modified version of clipping equation from [9]

[7] An efficient, directly physical model of this sub-circuit is actually fairly difficult to form, since the current flowing into the base of Q_{14} and Q_{15} cannot be neglected. Ignoring this current yields significant error in the cutoff frequency $f_c = 0.02$ and $G = {}^{R_{30}R_{63}}/_{(R_{30}R_{62}+R_{30}R_{63}+R_{62}R_{63})} = -24.12\,\text{dB}$. However, incorporating the current is non-trivial, due to the nonlinear dynamics of Q_{14} and Q_{15}.

6.2. Large-Signal

Figure 4 shows a short section of a SPICE simulation of the VCA. It is clear that the "upper edge" and "lower edge" bound the output signal and track the applied envelope signal V_{env}. These edges can be approximated very accurately as functions of V_{env}:

$$V_{\text{lowerEdge}} = \alpha_0 e^{-\beta_0 {V_{\text{env}}}^{\gamma_0}} + \alpha_1 e^{-\beta_1 {V_{\text{env}}}^{\gamma_1}} + \alpha_2 \quad (13)$$

$$V_{\text{upperEdge}} = \alpha_3 V_{\text{env}} + \alpha_4\,. \quad (14)$$

Curve-fitting of data tabulated from a SPICE simulation yields coefficients (with minimized error in the least-squares sense) for $V_{\text{lowerEdge}}$ and $V_{\text{upperEdge}}$ fits, which are shown in Table 1).

i	0	1	2	3	4
α_i	14.97	−0.01872	−14.33	1.011	−0.03122
β_i	0.01837	1.813			
γ_i	−0.3179	−2.988			

Table 1: $V_{lowerEdge}$ and $V_{upperEdge}$ fit coefficients.

Following this same procedure yields a model for the second VCA.

Now that the oscillators, biasing circuits, and VCAs have all been developed as behavioral models, we commute them together into a single rectangular wave oscillator. This oscillator has the duty cycle D and frequency f studied in §3, but a low output state of $V_{\text{lowerEdge}}$ and a high output state of $V_{\text{lowerEdge}}$.

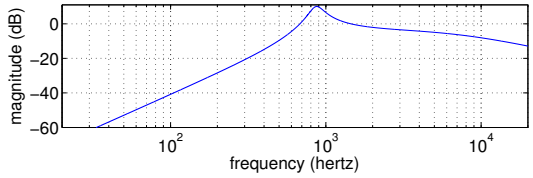

Fig. 5: Band pass filter magnitude response.

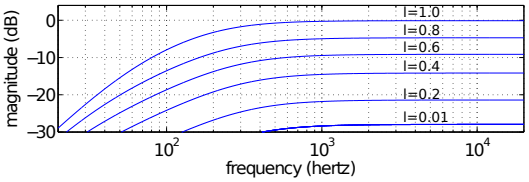

Fig. 6: Family of level control magnitude responses, with level control $l \in [0.0,1.0]$.

7. BAND PASS FILTER

The two outputs of the swing-type VCAs are applied to the two inputs of the band pass filter. It is inconvenient, though not impossible, to find the closed-form transfer function for this fourth order system by hand. An alternative approach, via Mason's gain formula, is shown in the Appendix.

Assuming linearity[8], the output V_{bp} of the band pass filter is the sum of the responses V_{bp1} and V_{bp2} to each of its two inputs. The continuous-time transfer function $H_{bp1}(s) = {}^{V_{bp1}(s)}/_{V_{\mathrm{VCA},1}(s)}$ characterizes the response of the system to the first input:

$$H_{bp1}(s) = \frac{\beta_3 s^3 + \beta_2 s^2}{\alpha_4 s^4 + \alpha_3 s^3 + \alpha_2 s^2 + \alpha_1 s + \alpha_0} \quad (15)$$

with coefficients:

$$\begin{aligned}
\beta_3 &= -R_{24}R_{25}R_{27}C_{28}C_{30}C_{31} \\
\beta_2 &= -R_{24}R_{25}C_{28}C_{30} \\
\alpha_4 &= R_{24}R_{25}R_{26}R_{27}C_{28}C_{29}C_{30}C_{31} \\
\alpha_3 &= R_{24}R_{25}R_{26}C_{28}C_{29}C_{30} + R_{24}R_{25}R_{27}C_{28}C_{29}C_{31} \\
&\quad + R_{24}R_{26}R_{27}C_{28}C_{30}C_{31} + R_{24}R_{26}R_{27}C_{29}C_{30}C_{31} \\
\alpha_2 &= R_{24}R_{25}C_{28}C_{29} + R_{24}R_{26}C_{28}C_{30} + R_{24}R_{26}C_{29}C_{30} \\
&\quad + R_{24}R_{26}C_{30}C_{31} + R_{24}R_{27}C_{28}C_{31} \\
&\quad + R_{24}R_{27}C_{29}C_{31} + R_{24}R_{27}C_{30}C_{31} \\
&\quad + R_{26}R_{27}C_{30}C_{31} \\
\alpha_1 &= R_{24}C_{28} + R_{24}C_{29} + R_{24}C_{30} + R_{24}C_{31} + R_{26}C_{30} \\
&\quad + R_{27}C_{31} \\
\alpha_0 &= 1
\end{aligned}$$

The magnitude response of $H_{bp1}(s)$ is found by evaluating $H_{bp1}(s)$ along the $s = j\omega$ axis and is shown in Fig. 5. Because of the symmetry of the band pass filter, $H_{bp2}(s) = {}^{V_{bp2}(s)}/_{V_{\mathrm{VCA},2}(s)}$ can be found by simply exchanging the terms R_{26} and C_{30} with R_{27} and C_{31} in Equation (15).[9]

[8] that the VCA outputs can be treated as ideal voltage sources, and that superposition holds

8. LEVEL STAGE

The output of the band pass filter is applied to the input of a level control stage, which offers control of the cowbell voice circuit's level. The output of the level control stage is applied to the non-inverting input of an op-amp configured as a unity gain buffer.

Assuming ideal op-amp behavior [10], the level control stage continuous-time transfer function $H_{le}(s) = {}^{V_{le}(s)}/_{V_{bp}(s)}$ is:

$$H_{le}(s) = \frac{\beta_2 s^2}{\alpha_2 s^2 + \alpha_1 s + \alpha_0} \quad (16)$$

with coefficients:

$$\begin{aligned}
\beta_2 &= C_{75}C_{76}R_{117}\mathrm{VR}_5 l \\
\alpha_2 &= C_{75}C_{76}\mathrm{VR}_5^2 l^2 - C_{75}C_{76}\mathrm{VR}_5^2 l \\
&\quad + C_{75}C_{76}R_{116}R_{117} - C_{75}C_{76}R_{117}\mathrm{VR}_5 \\
&\quad + C_{75}C_{76}R_{116}\mathrm{VR}_5 l + 2C_{75}C_{76}R_{117}\mathrm{VR}_5 l \\
\alpha_1 &= C_{75}R_{116} + C_{76}R_{117} - C_{75}\mathrm{VR}_5 \\
&\quad + 2C_{75}\mathrm{VR}_5 l + C_{76}\mathrm{VR}_5 l \\
\alpha_0 &= 1
\end{aligned}$$

where VR_5 is the maximum resistance of the level control knob with position $l \in [0.0,1.0]$.

Fig. 6 shows the magnitude response at various positions of the cowbell level control knob. To a first-order approximation, the level control stage is a passive voltage divider with two DC blocking stages.

[9] Since the stock components in the circuit have identical values ($R_{26} = R_{27}$ and $C_{30} = C_{31}$), the two transfer functions will be identical. However, they could differ significantly if new component values were swapped in.

[10] the output of the band pass filter acts as a perfect voltage source and no current flows into the non-inverting input of the buffer op-amp

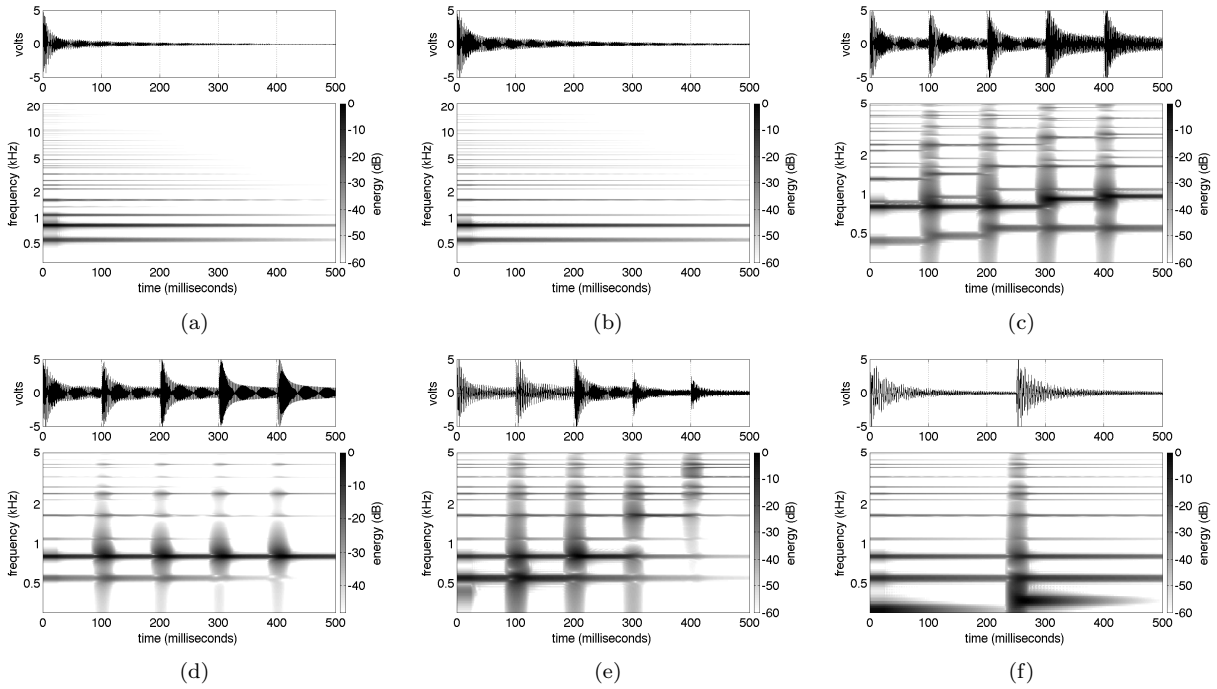

Fig. 7: Waveform/spectrogam pairs of cowbell voice circuit simulations: (a) a baseline SPICE simulation for comparison, (b) a baseline emulation via the physically-informed model, (c) changing rectangular wave oscillator fundamental frequencies via TM_1 and TM_2, (d) changing balance between oscillators by changing R_{26}, (e) changing band pass filter center frequency via R_{24} and R_{25}, and (f) setting high Q in band pass filter via R_{24} and R_{25}, to create a tunable decaying sinusoid at center frequency. All with $V_{ct} = 14$ V, $l = 1.0$, and sampling rate $f_s = 44100 * 8$ Hz.

9. MODELING

A digital model is implemented in `Matlab`. This model has the stock level control, as well as additional "bends" for tuning each oscillator over a wide range, adjusting the balance between the oscillators, and tuning the center frequency and Q of the band pass filter.

The trigger logic is implemented with simple algebraic manipulations on the accent level and timing data. All discrete-time filter coefficients are calculated via the bilinear transform. Since filter features are all relatively low with respect to typical audio sampling rates, warped bilinear transform methods are not necessary or implemented. Due to the potential bandwidth expansion in the VCA stage, some oversampling is used. Rectangular wave oscillators are implemented without alias-suppressed methods. Although not noticeable in the circuit's standard configuration (no bends), aliasing could be audible if the model's tuning was varied significantly. In this case, alias-suppressed methods[11] could improve the model.

Some parts of the implementation can introduce errors of their own. Using the Forward Euler Method in the envelope generator introduces extra delays between the VCAs and envelope generator. This could potentially be improved by implicit solving or an expanded model that combines both sections into one. However, since the envelope generator values change very slowly with respect to the sampling period, this extra delay should not introduce much error.

10. RESULTS AND CONCLUSIONS

Fig. 7(a) shows a reference SPICE simulation of the cowbell voice's standard configuration. Compare to

[11] for instance [10], [11]

Fig. 7(b), which shows the output of the physically-informed model. Fig. 7(a) and Fig. 7(b) show good agreement, which is supported by informal listening tests. However, there is not perfect agreement. Differences between the amplitude envelopes are evident in the waveform traces and differences between relative partial magnitudes are evident in the spectrograms. These errors are the result of simplified assumptions about the absence of loading between sub-circuits. In particular, the interface between the envelope generator, VCAs, and band pass filter is oversimplified in this physically-informed model.

Figs. 7(c)–7(f) show examples of "bends" based on component substitution, emulated with the physically informed model. Fig. 7(c) shows how oscillators 1 and 2 can be tuned via trimpots TM_1 and TM_2. Fig. 7(d) shows how the balance between the oscillators can be affected by changing R_{26} and R_{27}. Fig. 7(e) shows how the center frequency of the band pass filter can be swept by adjusting R_{24} and R_{25}. Fig. 7(f) shows how the band pass filter can be used as a third oscillator. Tuning the band pass filter's Q high enough will cause the ringing at the center frequency to last long enough to be perceived as a tunable third tone.

Audio examples and other supplementary materials can be found online at this work's companion page.[12]

11. ACKNOWLEDGMENTS

Thanks to Lauchlan "Locky" Casey for suggesting a TR-808 cowbell model as a follow-up to [2] and [3]. Thanks to Martin Kramer for his informative lecture on the use of Mason's gain formula for circuit analysis in CCRMA's Circuit Analysis and Modeling Seminar. Thanks to Melissa Kagen and Jing Chen for their help editing this manuscript.

APPENDIX: BAND PASS TRANSFER FUNCTION DERIVATION

It is desirable to find transfer functions in terms of the electrical component values of the circuit rather than finding them experimentally. This makes simulating the effect of mods based on component substitution trivial and exposes the transfer function's dependence on component values, supporting understanding of and modification to the original hardware.

[12] https://ccrma.stanford.edu/~kwerner/papers/AES14.html

However, finding a closed-form transfer function for a circuit with many reactances or many intermediate nodes can be prohibitively difficult by hand. This difficulty increases with filter order, and increases even faster with the number of equations needed for a full nodal or mesh analysis.

For the transfer functions in this work, a technique using Mason's gain formula was used to simplify calculations. Mason's gain formula simplifies the process of finding the transfer function of a linear signal-flow graph [12]. Its use is common in control theory, where dynamical systems are natively formulated in a signal-flow graph representation.

Mason's gain formula finds the graph gain G as the sum of products of forward path gains, loop gains, the circuit determinant Δ, and cofactors Δ_k [13]:

$$G = \frac{\sum_k G_k \Delta_k}{\Delta} \tag{17}$$

where

$$\begin{aligned} G_k &= \text{gain of the } k\text{th forward path} \\ \Delta &= 1 - \sum_m P_{m1} + \sum_m P_{m2} - \sum_m P_{m3} + \dots \\ P_{mr} &= \text{gain product of the } m\text{th possible} \\ &\quad \text{combination of } r \text{ nontouching loops} \\ \Delta_k &= \text{the value of } \Delta \text{ for that part of the graph} \\ &\quad \text{not touching the } k\text{th forward path} \end{aligned}$$

Linear electrical circuits do not natively have the properties of a signal-flow graph (a directed graph with branches representing multiplications and nodes representing system variables). Still, the system of equations describing their behavior can be rewritten as a signal-flow graph, allowing the application of Mason's gain formula.[13]

This technique is a very useful tool in physical modeling of analog circuits and virtual analog. To use this technique in this context, one can write out the set of linear equations describing the circuit's behavior (via KCL, for instance), then transform them into an equivalent signal-flow graph. The can be

[13] This is discussed for algebraic equations in general in [14]. [15] provides a review of Mason's gain rule and linear signal-flow graphs, and contains examples in the context of electrical circuits.

done easily by inspection if the individual (KCL, for instance) equations do not have many terms.

At this point, one can either apply Mason's gain formula manually, or automate it in software. This appendix will use Rob Walton's `mason.m`[14], which takes in a netlist-like file representing the signal-flow graph and gives results in a form that is convenient for further manipulation via `Matlab`'s symbolic manipulator.

This method has several advantages over direct algebraic manipulations of a system of equations. First, the most time-consuming part of the analysis can be completely automated. Second, due to this automation, any errors that are made in the initial stage of analysis cannot contaminate pages of algebra. Finally, this sort of analysis is useful as an alternate formulation that can reveal interesting dependencies.

This appendix shows a case study of how this technique can be used to find the transfer function of the band pass filter from §7.

First, start with KCL at nodes V'' and V_-:

$$\frac{V_{\mathrm{VCA},1} - V''}{R_{26} + \frac{1}{sC_{30}}} = \frac{V'' - V_{bp1}}{\frac{1}{sC_{29}}} + \frac{V''}{R_{27} + \frac{1}{sC_{31}}} + \frac{V''}{\frac{1}{sC_{28}}} + \frac{V''}{R_{24}} \tag{18}$$

$$\frac{V''}{\frac{1}{sC_{28}}} = -\frac{V_{bp1}}{R_{25}} \tag{19}$$

By introducing a few variable substitutions ($V_1 = V_{\mathrm{VCA},1} - V''$ and $V_2 = V'' - V_{bp1}$), expressions for each node voltage and substitution variable (except the input voltage $V_{\mathrm{VCA},1}$) result:

$$V_1 = V_{\mathrm{VCA},1} - V'' \tag{20}$$

$$V_{bp1} = V'' - V_2 \tag{21}$$

$$V_2 = V_1 \left(\frac{C_{30}}{C_{29}\left(sR_{26}C_{30}+1\right)} \right) - V'' \left(\frac{C_{31}}{C_{29}\left(sR_{27}C_{31}+1\right)} + \frac{C_{28}}{C_{29}} + \frac{1}{sR_{24}C_{29}} \right) \tag{22}$$

$$V'' = -\frac{1}{sR_{25}C_{28}} \tag{23}$$

[14] `http://www.mathworks.com/matlabcentral/fileexchange/22-mason-m`

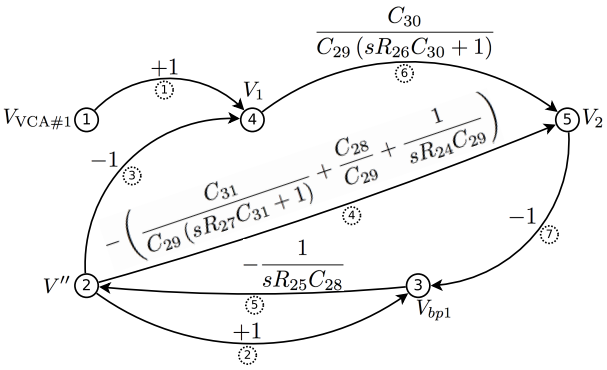

Fig. 8: Band pass filter algebra signal-flow graph.

Equations (20)–(23) can be transformed into a signal-flow graph by treating the left half of each equation as a dependent node and converting the algebra into path gains. The result of this transformation is the signal-flow graph shown in Fig. 8.[15]

To automate the application of Mason's gain formula, this signal-flow graph representation can be rewritten as a netlist that can be interpreted by `mason.m`.

Figure 9 shows the signal-flow graph in netlist form, using node and branch indices from Fig. 8. Each row in the netlist describes one branch in the signal-flow graph. The first column is a branch index, the second column is the start node index, the third column is the end node index, and the last column is the branch gain.

This netlist is referenced in a `Matlab` script, shown in Fig. 10, that calls `mason.m` and uses `Matlab`'s symbolic manipulator to rearrange the output as a transfer function (numerator and denominator are both polynomials in s). The results are shown in shown in Equation (15).

12. REFERENCES

[1] Roland Corporation, "TR-808 Service Notes, 1st Edition," June 1981.

[15] Note that this is only one of the many possible signal-flow diagrams that can describe Equations (18)–(19) In constructing a signal-flow graph, one must trade off the complexity of the path gains and the complexity of the overall graph, and find whatever balance is most convenient.

```
1 1 4 (1)
2 2 3 (1)
3 2 4 (-1)
4 2 5 (-(C31/(C29*(s*R27*C31+1))+C28/C29+1/(s*R24*C29)))
5 3 2 (-1/(s*R25*C28))
6 4 5 (C30/(C29*(s*R26*C30+1)))
7 5 3 (-1)
```

Fig. 9: `cowbellFilter.net`: band pass filter netlist for Mason's gain formula routine.

```
netfile = 'cowbellFilter.net';
[Numerator,Denominator] = mason(netfile,1,3); % input node 1, output node 3
Denominator=sym(Denominator);
Numerator=sym(Numerator);
syms s R24 R25 R26 R27 C28 C29 C30 C31;
collect(simplifyFraction(Numerator/Denominator),s)
```

Fig. 10: `Matlab` script to invoke `mason.m`.

[2] Kurt James Werner, Jonathan S. Abel, and Julius O. Smith, "A Physically-Informed, Circuit-Bendable, Digital Model of the Roland TR-808 Bass Drum Circuit," in *Proc. Int. Conf. on Digital Audio Effects*, Erlangen, Germany, Sept. 1–5, 2014.

[3] Kurt James Werner, Jonathan S. Abel, and Julius O. Smith, "The TR-808 Cymbal: A Physically-Informed, Circuit-Bendable, Digital Model," in *Proc. Joint Session of Int. Comput. Music Conf. and Sound and Music Computing Conf.*, Athens, Greece, Sept. 14–20, 2014.

[4] Robin Whittle, "Modifications for the Roland TR-808," [Online]. Available: `http://www.firstpr.com.au/rwi/tr-808/`, Oct. 7, 2012.

[5] Eric Archer, "Tr-808 Cowbell Clone w/ Mods," [Online]. Available: `http://youtu.be/ET4vkBluR_4`, 2009.

[6] Eric Archer, "TR-808 Cowbell Project," [Online]. Available: `http://ericarcher.net/devices/cowbell/`, 2009.

[7] Gordon Reid, "Synth Secrets: Synthesizing Cowbells & Claves," *Sound on Sound*, September 2002.

[8] Otto H. Schmitt, "A Thermionic Trigger," *Journal of Scientific Instruments*, vol. 15, pp. 24–26, Janary 1938.

[9] David T. Yeh, Jonathan S. Abel, and Julius O. Smith, "Simplified, Physically-Informed Models of Distortion and Overdrive Guitar Effect Pedals," in *Proc. Int. Conf. on Digital Audio Effects (DAFx-10)*, Bordeaux, France, Sept. 10–15, 2007.

[10] Timothy Scott Stilson and Julius O. Smith, "Alias-Free Digital Synthesis of Classic Analog Waveforms," 1996.

[11] Juhan Nam, Vesa Välimäki, Jonathan S. Abel, and Julius O. Smith, "Alias-Free Virtual Analog Oscillators Using a Feedback Delay Loop," in *Proc. Int. Conf. on Digital Audio Effects*, Como, Italy, September 1–4, 2009.

[12] Samuel J. Mason, "Topological Analysis of Linear Nonreciprocal Networks," in *Proc. of the IRE*, 1957, pp. 829–838.

[13] Samuel J. Mason, "Feedback Theory—Further Properties of Signal Flow Graphs," in *Proc. of the I.R.E.*, July 1956, vol. 44, pp. 920–926.

[14] C. L. Coates, "Flow-Graph Solutions of Linear Algebraic Equations," *I.R.E. Transactions on Circuit Theory*, vol. 6, no. 2, pp. 170–187, June. 1959.

[15] Martin Krämer, "Design of a CMOS Sample-and-Hold Amplifier for a High Precision Front-End Circuit Using an Extended g_m/I_{ds} Method," M.S. thesis, Technische Uinversität, Kaiserlautern, 2009.

View publication stats