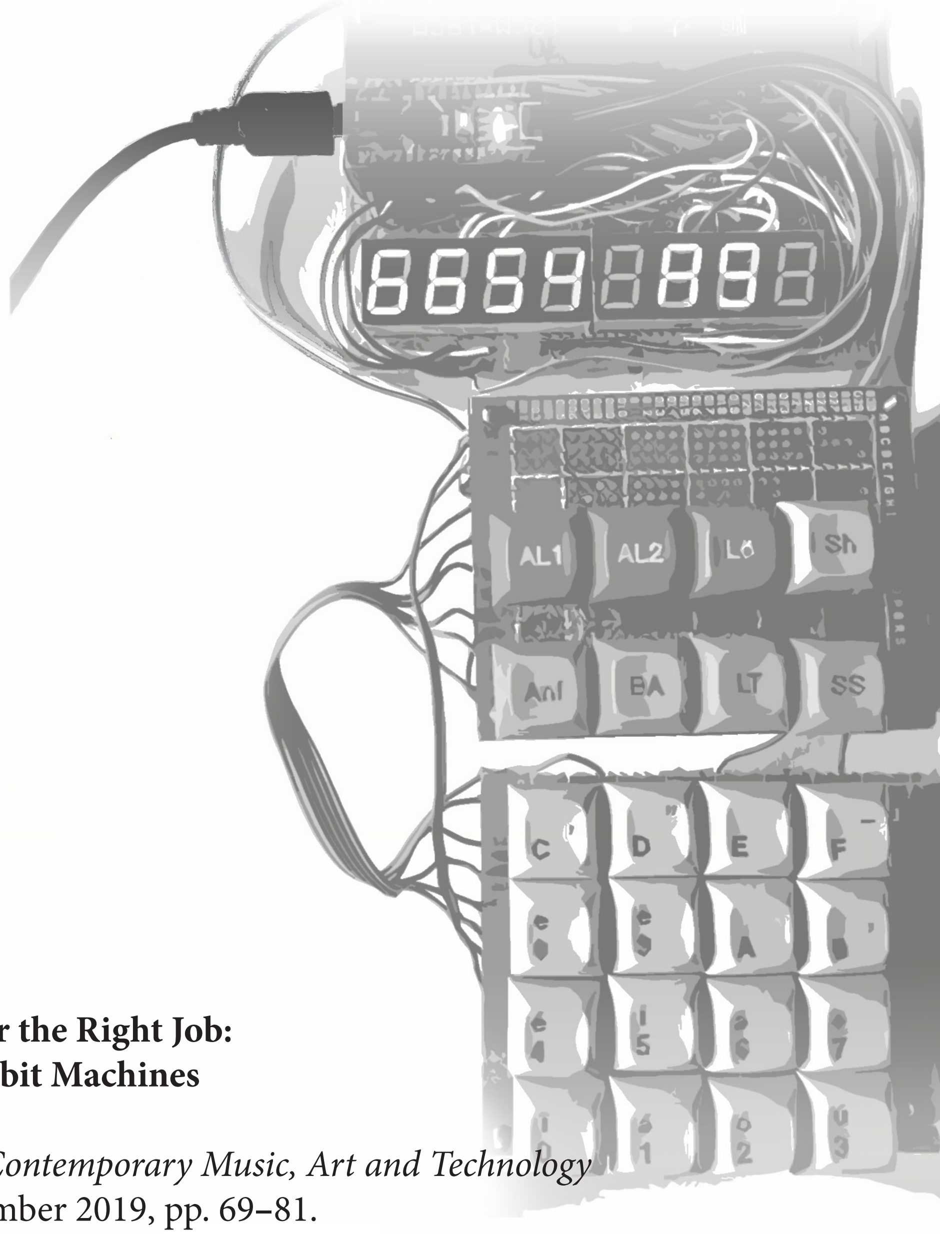
I N S Δ M

JOURNAL OF CONTEMPORARY MUSIC, ART AND TECHNOLOGY



The Wrong Tool for the Right Job: Composition on 8-bit Machines

Tobias Banks

INSAM Journal of Contemporary Music, Art and Technology
No. 3, Vol. II, December 2019, pp. 69–81.

I N S Δ M

UDC 78.02: [794:004.42

Tobias Banks*
California State University Long Beach
Long Beach, California, United States

THE WRONG TOOL FOR THE RIGHT JOB: COMPOSITION ON 8-BIT MACHINES

Abstract: An artist's medium can inspire them to reach new heights with its possibilities or curtail their ambitions with its limitations—but, ultimately, it is what gives shape to their artistic vision. Few artistic mediums exemplify the conflict between possibilities and limitations, and the sheer ingenuity of artists as they balance these two forces, as well as computer music does. This paper will delve into the rich tradition of demoscene and videogame music, that flourished in the early days of personal computers and gaming consoles, to examine how the hardware used by different composers affected their processes. This paper discusses the technical specs and rich library of music of three very different pieces of hardware: the Commodore 64, the ZX Spectrum, and the Nintendo Entertainment System. The C64, with its powerful and revolutionary "SID chip" sound card, demonstrates the amazing potential computer music offers to those determined enough to surmount the technical challenges. The ZX Spectrum, with its one-channel beeper speaker, shows how a resourceful artist can wring brilliance from even the most limited of mediums. And the NES, with its five channels of pure nostalgia, teaches us that a couple of square waves and some heart can inspire a generation.

Keywords: computer music, synthesizers, demoscene, videogame music, 8-bit, C64, NES, ZX Spectrum, Nintendo, chiptune

In the 2008 documentary "It Might Get Loud," Jack White of the White Stripes describes his creative process: "I keep guitars that are, you know the neck's bent and it's a little bit out of tune. I want to work and battle it and conquer and make it express whatever attitude I have at the moment. I want it to be a struggle." White derives his inspiration from the limitations and obstacles in his path. And when

* Author's contact information: tobiasonguitar@gmail.com

there aren't enough roadblocks, he goes as far as to create them himself. Necessity is the mother of invention after all, and some of the most transcendent moments in music arose from seemingly insurmountable limitations. Ravel's *Piano Concerto for the Left Hand* was written for pianist Paul Wittgenstein after he lost his right arm in the First World War. The piece stands not only as a brilliant testimony to human resilience and ingenuity, but also as one of the most unique and creative pieces in Ravel's *oeuvre*. Electronic music has historically weathered criticism from the uninitiated as the antithesis of this kind of ethic. Many music listeners believe that drum machines and sequencers make music-making too easy and remove the kind of struggle that leads to real inspiration. But the most influential developments in the genre have often arisen from the direst circumstances. Stockhausen repurposing telephone test equipment for his great opuses is one example. And then there are the Jamaican dub producers and sound system DJs who had limited access to live musicians and started experimenting with electronics to cut costs. Using nothing but four-track mixers and whatever equipment they could build themselves, these innovative producers rocked the streets of Kingston with spacey sounds, on an extremely tight budget. The best electronic composers are resourceful and know how to turn limitation into inspiration. A prime example of this struggle between the composer and his tools is the genre of early videogame music.

Composers in the 8-bit era faced myriad technical and logistical challenges unique to their time. On the one hand, they had to contend with crushing deadlines, storage limitations, and the chaos and uncertainty of working in the fast-moving tech industry. And on the other hand, they were essentially creating a new musical genre from scratch using unfamiliar tools. How do you write an engaging composition that only lasts thirty seconds or so using only a few square waves? It hadn't really been done before. Faced with these obstacles, the talented game programmers/composers of the 80s and early 90s bent their hardware to their will to create otherworldly sounds, and crammed what little storage space they had with timeless melodies. Of this hardware, three machines stand out as being particularly inspiring sparring partners for dedicated programmers. The NES/Famicom—with its massive library of 1000+ games and its meat-and-potatoes sound chip, comprised of two square waves, a triangle wave, a noise channel, and a rarely used PCM sampling channel—shows how different composers can take a small collection of sounds and timbres and create music in hundreds of distinct styles. The C64, with its powerful MOS Technology SID chip—which boasts four waveforms, smooth pulse-width modulation, filters, and some great ring modulation capabilities—became a proving ground for the most inventive game composers and demoscene wizards. And the ZX Spectrum, the UK's budget personal computer—with its single channel, 1-bit beeper—was the ultimate technical challenge, inspiring composers to feats of extraordinary programming and compositional creativity. The limitations of these different machines meshed and collided with the artistic ambitions of composers, the narrative demands of the games themselves, and the aesthetic proclivities of videogame fans, producers, and developers, to create a vibrant and diverse tradition of chiptune music.

Technical challenges weren't the only driving force of the chiptune aesthetic, although they are the main focus of this paper. As if creating large quantities of music on recently invented digital instruments wasn't enough of a struggle, composers also had to contend with editorial mandates from publishers, the tonal, structural, and emotional demands of the games themselves, and write the music they wanted to write, all while trying to appeal to the general gaming public. Much can be said for the different work cultures surrounding the three systems. NES games were often blockbuster commercial products made by large teams. With the C64 and ZX Spectrum, however, games were smaller affairs, involving a few programmers and a rotating stable of composers, drawn from the few musicians in the area who were good enough programmers to coax music from these early personal computers. These social factors are unique to this specific sector of the entertainment industry and they are a strong influence on the development of the game music genre.

NES

The NES sound chip was designed with one purpose in mind: to provide impressive in-game music as efficiently as possible. This meant that the chip had to make it easy for composers to write music in one relatively limited style, while ensuring that the sounds used as little storage space as possible. Despite this, music on the NES represents a diverse array of different styles and genres, from the groovy Dixieland jazz of Super Mario Bros to the dark and theatrical classical/rock fusion of Castlevania. By looking at the technical limitations and capabilities of the NES chip, we can gain insight into how it shaped the music of the NES, and how composers were able to twist and exploit this technical framework to suit their own needs.

The vanilla NES/Famicom has five channels, although expansions like the Famicom Disk System and various in-cartridge expansion chips such as the Konami VRC6 and 7 chips added additional audio capabilities. The first two channels are pulse waves with variable duty cycles, which can be set to 12.5%, 25%, 50% (square wave), or 75%. These are the NES's two "lead voices," and it's quite remarkable that these two channels carried so many iconic tunes. The only expressive options they provide are the slightly different timbres afforded by the variable duty cycles, envelope generator, and modest vibrato controls. NES games typically required the soundtrack to address the theme, tone, and narrative of the game with genre-specific music, so having a monochromatic beep as the lead voice could conceivably present some major challenges. But in the hands of capable composers, these square waves became the heroic brass section driving Link onwards on his journey through Hyrule in The Legend of Zelda, the haunting organ reverberating through Dracula's castle in Castlevania, and the driving guitar power chords that gave "Rockman" (the original Japanese name of "Mega Man") his name. The third channel is a triangle wave that's usually used for basslines, and also, by the use of sharp downward glissandos, as a kick drum. Next is the noise channel which is used in conjunction with the triangle wave, and occasionally, along with the fifth channel, for percussion. Finally we have the mysterious and rarely used fifth channel for PCM sampling. It is somewhat underutilized, because samples take up a lot of memory, and it seldom made sense to sacrifice entire levels of gameplay for some horribly bitcrushed cowbell sound effect. But those brave enough to learn its secrets found a lot of clever uses for it, as we shall see.

One of the first challenges a NES composer faces is an orchestration one: You only have three voices. There are a few "standard" textures that you'll see in the majority of soundtracks for the system. The most basic approach is to have a simple triangle wave bassline outlining the chord progression in the bottom, alongside an arpeggiated or alberti bass square wave that serves as the main harmonic accompaniment, with the melody on top of that. This rather pianistic style can be found at its most refined in Nobuo Uematsu's Final Fantasy score. Another common approach is to have a driving bassline supporting the two square waves as they pound out the melody in fifths or thirds, occasionally breaking off from each other in little countermelodies, as in the main theme for Castlevania. And with game series like Castlevania and Dragon Quest, where the music has a decidedly classical vibe—Dragon Quest because it is an epic RPG scored with fantasy-orchestral music, and Castlevania because it has its own brand of scary-organ-rock—composers have the luxury of being able to draw from various classical styles as well. A driving fugue, for instance, makes very economical use of the NES's three voices to create a rich texture. But while classical styles and textures work perfectly for boss fights with Dracula, it might have been a little stylistically jarring for a game like Bad Dudes. So how does a composer create a more modern texture?

Most game scores try to find ways to adapt the NES's five voices to create digital "ensembles" that fit their desired style. But a few interesting works attempt to distort the listener's perception of continuous "voices" to create a denser, more orchestral texture in which it isn't always clear which voice is doing what. Enter: Mega Man. The Mega Man series was actually scored by a few different composers over the course of six games on the system, but the formula introduced by the first game's composer, Manami Matsumae, is pretty consistent throughout all of the series' NES entries. Through intense syncopation, rhythmic inventiveness, and clever counterpoint, Mega Man composers were able to create rich polyphonic textures and compound melodies that make it very difficult to isolate the three voices. This density creates the illusion of a full rock band and helps the listener imagine the music as more than just a few voices of bleeps and bloops. The track "Metal Man," from Mega Man 2, is a musical tapestry of interlocking parts that drift in and out of clarity, sometimes breaking into distinct melodies, occasionally joining together into cathartic power-chord blasts—but usually bouncing off one another in a web off controlled chaos, held loosely together by some repeated dotted eight + sixteenth note motives. The marching drumbeat is the only reliable point of stability as the two square wave channels dance around each other, and the shaky bassline lays out undulating grooves. The Mega Man series is so committed to dense polyphony that tracks like "Quick Man" from Mega Man 2 sacrifice percussion so that the noise channel can be freed up to create a pitched, squawky noise that serves as an extra voice of polyphony.[2] The Mega Man series score doesn't sound like much else, but it does call to mind the contrapuntal guitar pyrotechnics of bands like Television and Quicksilver Messenger Service. It's a very unique style that was inspired by the composers' desire to squeeze as much instrumental power out of the NES system's five channels as possible.

Another drawback of the NES is its severely limited range of timbres. One of the few ways composers can add some timbral variety to their tracks is through the variable duty cycle on pulse waves. With only four steps (or rather three, as the 75% and 25% options are inverted and sound the same), it is untenable for continuous pulse-width modulation. The most obvious application of the variable duty cycle is to add timbral variety to orchestrations. The 12.5% setting has a distinct nasally tone, and with some good orchestration and a little bit of imagination it can create a "brass section" type sound. See how the short fanfare at the beginning of Tim Follin's title screen theme for Solstice conjures up images of heralds announcing the king's arrival with bugles. Composers use the 25/75% option fairly often, as it has more harmonics than the square, missing only every fourth harmonic, rather than every second. The 12.5% is even better, missing only every eighth, and it is used less often and usually in conjunction with the 25%/75% options. But the duty cycles are useful for more than just some slight timbral variety. One common trick is to change duty cycles at the attack of a note to create a pizzicato-type effect. This can be heard prominently in Gremlins 2's "Wise Man's Shop Theme," where it's used to create a pipa-type effect in the accompaniment, to fit the Chinatown curio shop.

Another interesting use of the duty cycle is demonstrated by Jeroen Tel's score for Alien 3. Tel opted for a heavy metal themed score, which fits the aesthetic of the game. He achieves all sorts of heavy metal guitar effects by manipulating the duty cycle. By changing the duty cycle about a sixteenth note into a note, Tel emulates the sound of a guitarist who makes heavy use of artificial harmonics, such as Zakk Wylde. This trick is used on pretty much every track, but the guitar effect is probably at its most recognizable at the beginning of the "1st and 3rd Guardians" level. Finally, another very common use/abuse of the duty cycle is to just switch through extremely fast to create a very videogame-like tremolo freak-out effect. This trick is used in pretty much every other bar in Alien 3 to suggest fast guitar shredding, but you can find it in most other NES games—if not as a musical element, then as a sound effect.

Another feature of the NES that's ripe for exploitation is the mysterious PCM channel. One of the most well-known applications of this channel is for digitized

2 The noise channel trick can be observed in this remake created using "Famitracker," a tool for creating NES music on modern systems: https://www.youtube.com/watch?v=TqThwH9eMyo.

speech. Many young gamers were traumatized by the oft-heard "game over" in Top Gun. And who could forget the ominous threat of "Ski or Die" in the title screen of the eponymous classic? Or the blood-curdling death-howl of men plummeting to their Styrofoam deaths in American Gladiators, the videogame based on the game show? But these gimmicky audio snippets pale in complexity to the technical marvels of the Dirty Harry game, which features the entire "Feelin' lucky, punk?" speech in digitized form, alongside an animated cutscene. But of course, the sampling channel can also be an impressive musical tool, provided you have the storage space. It is most commonly used for adding percussion. Just one sampled snare or kick can go a long way towards beefing up a game's sound. Tim and Geoff Follin's supercharged Silver Surfer score is an excellent example of this.

The channel also has a pitch shifter, which allows composers to use the PCM channel melodically. This feature was used most famously to create the "Sunsoft bass" sound, which was named after the publisher that featured it in many of their games. When scoring games like Batman and Journey to Silius, Sunsoft composers would take a sample of a more complex waveform than the NES was capable of creating, and then pitch shift it to create basslines that sounded like nothing else on the market. The capabilities of the sampling channel are pretty much only limited by storage space, so while it was never used to its full potential during the system's life, today's chiptune composers, freed of storage limitations, can use it to create music that sounds like it came from a completely different system.

Over the NES's life cycle, programmers devised many graphical and gameplay improvements as they learned how to extract the system's full potential. But the musical advancements are the most impressive. Take the score for a launch title like Super Mario Bros, where you'll hear a lot of timeless melodies that keep you coming back for one more level. But, just a few years later, chiptune would be developed into an art form by composers like Tim Follin, who took the system's limitations as a challenge. Follin's title screen for Solstice begins with a modest and clumsy two-voice canonic fanfare, played on square waves at a nasally 12.5% duty cycle for comedic effect and to simulate brass instruments. After convincing you that this is just a fun puzzle game—possibly an old arcade or PC port—he hits you with all voices on max volume. Swirling arpeggios and duty cycle runs suggest a Rick Wakeman synthesizer solo. Sampled drums assault your senses before tapering away into a cascade of simulated delay, and a court jester playing the recorder rises to the fore with a lively prelude. Soon a full medieval consort, emulated through clever use of envelope, duty cycle, and vibrato, breaks into a lively gigue, accompanied by a driving prog rock rhythm section occasionally supported by synthesizer effects. Music on the NES had developed so far, technically, that a composer could create dramatic effect by referencing more primitive scores from earlier in the system's life and then subverting the listener's expectations by unleashing the machine's full capabilities.

C64

Bob Yannes (2014), the creator of the C64's SID chip, said this of his work: "I thought the sound chips on the market (including those in the Atari computers) were primitive and obviously had been designed by people who knew nothing about music… I was attempting to create a synthesizer chip which could be used in professional synthesizers." These plans would never come to fruition. Yannes admits that he had been in talks with Sequential Circuits, who were interested in possibly buying the chip, but this never amounted to anything. But, although it didn't find any use on stage or in the recording studio, Yannes had invented one of the most advanced electronic instruments of his time, and a community of dedicated composers and programmers continues to discover new capabilities on it today.

The C64 only has three channels, but unlike the NES, these channels can be used for anything. This three-channel setup creates a drastically different environment for composers than on the NES. On the NES, each channel is dedicated to and optimized for a predetermined function: squares for leads, triangle for bass, noise for percussion, and a limited audio playback channel for sound effects. So in essence, every NES soundtrack is played by the same "house band," and composers are encouraged to play to that band's strengths. But the C64, with its three channels that can all create any sound at any given time, changes the composer's mindset. In this type of environment, experimentation is the name of the game. Composers are pushed to tinker with different "ensembles" to try to make more efficient use of their three channels of audio real estate.[3] Clever programming, exploits, and shortcuts become the domain of any C64 composer who wants to make an impression. So a simple C64 track might have the bassline in one channel, fast arpeggios and effects in another, a continuous melody in the last, and all kinds of percussion sounds spread across all channels wherever they find space. But that is really just the tip of the iceberg. With a bit of clever orchestration, a composer can achieve any kind of ensemble they can dream up.

On top of this, the range of sounds available with the SID chip are comparable to analog synths of the day. It has pulse waves with near-analog pulse-width modulation. Throbbing PWM would become a staple of the C64 catalog, making for good simulated "guitar solos" and acid basslines. This electrifying sound immediately screams "SID" to any retro gamer. This was accompanied by a triangle wave and a saw, which was rare in 8-bit machines, and this lent a lot of timbral depth to the system and set it apart from the more "bleep-bloopy" sound world usually associated with 8-bit machines. And of course, there was a noise generator for percussion. The Commodore also comes with amplitude and ring modulation, and unlike any other

3 There are a few channels on YouTube that post oscilloscope views of C64 tracks so you can see the interplay between the three tracks. The two tracks discussed in this paper can be viewed here:
https://www.youtube.com/watch?v=pgPEaI0GHBI.
https://www.youtube.com/watch?v=26zxf8vXIs0.

system of the time, analog filters. This was virtually unheard of at the time, and these features really place the C64 alongside hardware synths of the era.

Since the Commodore is such a powerhouse, there is a trend in C64 music that isn't present in other 8-bit traditions: composers imposing artificial limitations on themselves for extra challenge. One of the more hauntingly beautiful tunes created on the system is Jereon Tel and LMan's "$11 heaven," named after the waveform ID for the SID's triangle wave. As the name suggests, the piece uses nothing but triangle waves throughout its three-minute duration. But through subtle envelope programming, some sound masking, and good songwriting, the composers craft a soothing, ethereal soundscape full of rich, textured pizzicati; sweet, innocent melodies that drift in and out like daydreams and reveries; and bouncy, groovy basslines. Since the triangle wave is somewhat underutilized in favor of the more dynamic pulse and saw waves, this tune is a novelty in the Commodore catalog. Had I first heard it without any context, I might not have even recognized it as a C64 tune. While the composers are limiting themselves to a small portion of their machine's capabilities, it is worth noting that a texture this simplistic wouldn't have been possible on the NES, which only allows one triangle wave at a time due to its predetermined channel roles. "Less is more" is generally not an applicable axiom in the world of chiptune, where composers usually want to squeeze the most out of the few tools they possess. But the SID chip is so advanced that it affords composers the luxury of practicing restraint.

On the maximalist end of the spectrum, we have composers like Rob Hubbard. Hubbard is probably the most famous SID composer. He is known for pushing the system to its limits to achieve as many perceived "voices" as possible and emulate a wide variety of instruments. He pioneered all kinds of technical innovations in SID programming to achieve a distinct sound on the Commodore, that sets it apart from all the other machines of its era. One of his most famous works is his music for the arcade conversion of Commando. Hubbard says of his experience writing the score: "There is an interesting story behind Commando. I went down to their office and started working on it late at night, and worked on it through the night. I took one listen to the original arcade version and started working on the C64 version. I think they wanted some resemblance to the arcade version, but I just did what I wanted to do. By the time everyone arrived at 8 a.m. in the morning, I had loaded the main tune on every C64 in the building! I got my cheque and was on a train home by 10 a.m." (Warren 2019). Most will agree that this overnight job came out sounding far superior to the original arcade version. The arcade cabinet uses an advanced Yamaha FM sound chip based on the DX7 synth, but while this technology is arguably more advanced than the SID chip, the original arrangement is somewhat lackluster. It's a simple driving melody over a marching snare drum, to let you know it is a war game. The chip's eight channels are mostly used up playing basic synth chords. Hubbard's version, on the other hand, sounds more like a rock band backed up with a full orchestra, despite the fact that he only has three channels to work with. In any given measure you can hear at least five distinct voices, but over the course of the track he cycles through different sounds, giving the impression of a very large ensemble. This is all accompanied by aggressive breaks from a very strange, glitchy percussion section that is a staple of Hubbard's flashy, dynamic sound.

ZX Spectrum

Compared to both the NES and the C64, the ZX Spectrum is technically the most limiting, as it was never really meant for music-making. It is really a miracle that anyone ever managed to get more than short, monophonic melodies out of it. But paradoxically, in some ways it is less creatively limiting than either the NES or Commodore. We looked at how the NES sound chip was designed to emulate one "ensemble," and how that stifles innovation and limits composers by encouraging them to stick to the strengths of the NES "house band." And we compared this workflow to that of the Commodore, which functions more like a normal synthesizer, and allows any combination of sounds as long as there are no more than three of them happening at the same instant. But these considerations are nonexistent with the ZX, because the ZX Spectrum has no sound chip. The Spectrum was very affordable, launching with a price of £125 for the 16 KB RAM model and £175 for the 48 KB model, and these prices would soon drop even lower. For reference, the C64 launched at £399 in the UK. So at that incredible price point, the ZX came equipped with what you might call limited sound capabilities: A single 1-bit "beeper" speaker that is addressed directly by the CPU. This meant that all sounds on the Spectrum were essentially created from scratch, with 1s and 0s, and that you had to be an actual wizard to make music on it.

Originally the beeper was used to produce simple square wave tunes, and that is likely all the designers ever imagined for it. And although the phenomenon of music on the "Speccy" would evolve into something much stranger and more complex, it is worth examining how programmers utilized the speaker when it was seen only as an added gimmick. There was very little original music written in this format. Most of the games from this period were written by one or two programmers, who usually weren't musicians. Since it is an impossible task to create a coherent and effective musical statement using only a single square wave in the space of about 20 seconds, it makes more sense to borrow tunes that already have meaning attached to them. A good example of this is the score for the classic game Jet Set Willy. Jet Set Willy is a sequel to Manic Miner, and it sees series protagonist Miner Willy moving into a haunted house with the loot he found in the previous installment of the series. The title screen opens with a few bars of the somber left-hand figure from Beethoven's Moonlight Sonata, rendered as a series of expressionless bleeps. The musical reference immediately communicates the horror theme, and the fact that it is played so crudely through the ZX speaker adds a layer of comedic parody that suits the tone of the game. This could not have been achieved with an original melody.

The in-game soundtrack is also an effective choice for a tense platform game. There is, for example, an even bleepier rendition of Grieg's cartoon soundtrack staple "In the Hall of the Mountain King," which provides some excitement and whimsically spooky vibes.

It is remarkable that some ZX Spectrum games have in-game scores and sound effects. This really shouldn't be possible because, as I mentioned, the Spectrum uses the CPU to create all its music, so it normally wouldn't be able to handle music, graphics, and gameplay all at the same time. But with some clever programming, you can actually hear some sounds during gameplay in the form of short-pitched beeps. Maintaining a continuous pitch eats up runtime that could otherwise be used for gameplay, but by creating the shortest possible sounds, programmers can deploy only the minimum amount of sound necessary to create a desired effect. These small clicks are achieved through a technique known as "impulse trains." By limiting the duty cycle of a sound to a single bit, you can cause a speaker to pop. Kenneth B. McAlpine (2015) explains this process:

We could… reduce the number of ones in each cycle of the wave to create smaller and smaller duty cycles, varying the frequency spectrum and tone of the sound, until we send the beeper just a single positive bit, followed by a stream of zeroes. This signal is a binary impulse, and its Fourier transform is a constant. In other words, an impulse contains all possible frequencies in equal strength. It is not possible to hear an impulse on its own, but it is possible to hear the effect on the speaker of trying to play an impulse, the so-called impulse response. Any speaker exhibits a degree of inertia, taking a short but finite time to move from rest to maximum displacement and back again, and it's this response that can be heard as a noticeable click. Sequence a series of binary impulses together, separated by short gaps, and the result is an impulse train, a pitched tone whose frequency is determined by period between successive impulses, and which contains all of the harmonics of the signal at equal strength.

McAlpine discusses the specifics of generating certain pitches in his paper. Manic Miner uses these impulse trains to create both impressive music and sound effects. The first four bars of "In the Hall of the Mountain King" are hammered out mercilessly in a series of aggressive blips. This is carefully interlaced with a very soothing jumping effect that sounds like a babbling brook in comparison. These are both placed so that neither routine interferes with the other. This style of sound can be found in a few Speccy titles, and as long as you don't wear headphones, the effect can be very charming. These soundtracks are reminiscent of works by Ryoji Ikeda, in that they are composed of some of the most grating and painful sounds that computers are capable of making, played in an endless repetition—and yet, once your ears acclimatize to the harsh sounds, it's very mesmerizing and Zen-like.

As we've learned from the pieces we've discussed, pulse-width modulation is a powerful tool in computer-based music. Smooth analog waveforms, filters, and all the methods discovered by electrical engineers to coax sound from circuits, don't transfer easily to the digital world (although, the SID chip does a good job). But a pulse wave is just an on/off switch, and computers are built on 1s and 0s. Since the ZX can only generate pulse waves, ZX composers, unspoiled by the luxuries of a sound chip, explored the possibilities of duty cycles on a massive scale. You can hear timbral pulse-width modulation in plenty of simple monophonic tunes. Tim Follin's very first credit for Subterranean Stryker is a fine example of a catchy tune given a bit of life through PWM. It would be Follin who would eventually show us that there are much more impressive applications for PWM. It isn't an uncommon technique in traditional synthesis to speed the pulse-width modulation on a square wave up into audio range to achieve some FM timbres. And with some tweaking, you can actually create two clear, distinct pitches. This is a good trick in analog synthesis , but for a ZX composer with a solitary pulse wave to work with, it is everything. It is the key to polyphony.

Speccy composers essentially built all of their sound by controlling the series of 1s and 0s that make up a pulse wave. It's really the only thing the speaker would allow them to do. But in the hands of an ambitious programmer, this fine control of one waveform is just enough control to get a foot in the door when it comes to polyphony. Just as they had done with impulse trains, programmers used PWM to get their speakers working overtime. McAlpine (2017) explains:

As discussed earlier, sending different sequences of ones and zeroes to the beeper allows the creation of a series of related wave shapes, from trains of binary impulses through to pulse waves of varying duty cycle. This idea can be taken one step further by returning to the idea of speaker inertia, which is the notion that a speaker cone cannot change its state discretely and instantaneously. When driven, it takes a short but finite time to reach maximum displacement and must move through all its intermediate states between fully off and fully on. The speaker behaves in a similar, though not identical way, as it returns to rest. Modulating the width of the signals (by varying the amount of time that the speaker is driven relative to the time that it is not) sent to the beeper, the speaker can be driven to intermediate points between off and on, thereby simulating the effect of a continuous analogue voltage.

Zombie Zombie was the first game to use PWM to achieve polyphony, and right away the possibilities were apparent. But as hinted earlier, polyphony on the Spectrum is heavily associated with one man: Tim Follin. Follin took these techniques to new levels, reaching up to five voices on tracks such as Agent X. These tracks all have such a distinctive sound that it's fair to say that he created his own genre of music on the ZX. The grainy, distorted sound that comes from this technique adds a rich, shoegazing-like lo-fi aesthetic to Follin's prog-influenced harmonies, and the result is a psychedelic mind-trip like nothing else. The process of faux polyphony on the spectrum creates so many harmonics that it must have sounded comically cheap at the time. But to modern audiences it can actually sound a lot more complex and richer than the NES, and at times even the C64. The full, distorted chords of the ZX sound positively human, and bring to mind the harmonizer-tinged guitars of post-punk power trio Hüsker Dü. His opening for Chronos has a funky bassline, rich Juno-esque pads, and slow David Gilmour glissandi at the Spectrum's one volume, fundamentally altering your brain chemistry. These songs represent a style of synthesis that hasn't really been explored anywhere else. And no one set out to create this sound—it arose out of pure necessity. It exploded, in a burst of pure inspiration, from what is quite possibly the most limited format on which anyone has ever been forced to create music.

List of References

Bakke, Rolf R. YouTube. YouTube, February 14, 2017. https://www.youtube.com/watch?v=pgPEaI0GHBI.

Bakke, Rolf R. YouTube. YouTube, May 7, 2019. https://www.youtube.com/watch?v=26zxf8vXIs0.

Carne, Zedicon. "FamiTracker - Mega Man 2: Quick Man Stage Theme (2A03 Cover)." YouTube. YouTube, February 14, 2013. https://www.youtube.com/watch?v=TqThwH9eMyo.

"Interview with Bob Yannes: SID Chip Inventor." AUSRETROGAMER, February 17, 2014. http://www.ausretrogamer.com/interview-with-bob-yannes-sid-chip-inventor/.

It Might Get Loud. *Sony Pictures Classics*, 2009.

McAlpine, Kenneth B. 2015. "All Aboard the Impulse Train: A Retrospective Analysis of the Two-Channel Title Music Routine in Manic Miner." The Computer Games Journal 4, no. 3-4: 155–68. https://doi.org/10.1007/s40869-015-0012-x.

McAlpine, Kenneth B. 2017. "The Sound of 1-Bit: Technical Constraint and Musical Creativity on the 48k Sinclair ZX Spectrum." The Italian Journal of Game Studies 1, no. 6.

Pilkington, Warren, and "Kenz". Interview with Rob Hubbard. Accessed September 9, 2019. http://www.sidmusic.org/sid/rhubbard.html.

THE WRONG TOOL FOR THE RIGHT JOB: COMPOSITION ON 8-BIT MACHINES
(summary)

This article examines how the technical challenges presented by the Nintendo Entertainment System, the Commodore 64, and the ZX Spectrum affected the way composers wrote for those platforms. The NES, designed as a commercial game-machine, offered composers a narrow but reliable range of sounds with which to construct adrenaline-pumping tunes. The Commodore 64 was designed to emulate higher-end hardware synths and serves as a veritable playground for inventive computer musicians. And the ZX Spectrum, which was never intended for music making, was the ultimate test of a composer's programming chops. This piece also explores the culture of game developers and gaming fans and how this factor shaped the music of early gaming.

Article received: September 16, 2019
Article accepted: October 15, 2019
Original scientific paper