256B on the C64 for 8K

Nick Montfort
Massachusetts Institute of Technology
Cambridge, MA 02139, USA
nickm@nickm.com

C&C '19, June 23–26, 2019, San Diego, CA, USA
© 2019 Copyright is held by the owner/author(s).
ACM ISBN 978-1-4503-5917-7/19/06.
https://doi.org/10.1145/3325480.3329174

Abstract

A tiny program for the big screen; a classic computer driving the latest, cinematic display. My contribution to the Art Exhibition for Creativity & Cognition / Design Interactive Systems 2019 is a 256-byte executable for the Commodore 64, written in 6502 assembly, for the auditorium’s 8K display. This non-interactive program will produce graphical effects using only the character-based facilities of the Commodore 64’s VIC-II and producing sound via the system’s SID. I describe its relationship to my practice and concrete poetry, retrocomputing, sizecoding, the demoscene, and platform studies. The program’s output is suitable for presentation as a “short” with other short experimental motion pictures/executable artworks or prior to one or more longer works. Bridging the border of time and different eras of computing, the program will be run on an original hardware NTSC Commodore 64 (not in emulation) with the video output upscaled.

Author Keywords

Concrete poetry; visual poetry; non-interactive; demoscene; sizecoding; retrocomputing; Commodore 64; platform studies and practices; computer history.

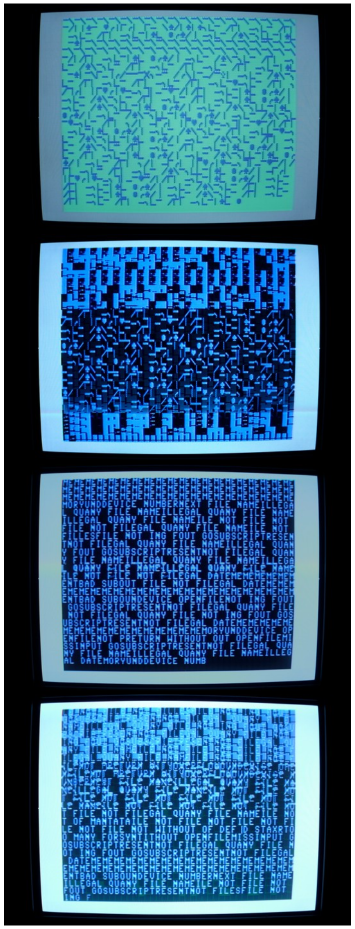


Figure 1: *Traumaphore* (2017), a 64-byte C64 program I wrote in 6502 assembly. Among other things it generates text using a bigram per-character Markov chain using BASIC error messages from ROM.

Introduction

What is provided here is context for this project — a non-interactive, text-mode, tiny program running on a computer from more than 35 years ago for projection on a state-of-the-art display — instead of documentation. The program itself will be made available for anyone to download and run on their Commodore 64 or in an emulator, but what lies behind and around the project can be harder to access. Rather than mention my most directly relevant projects in this main text, I have placed photographic documentation of some of my most relevant work in sidebars. The *256B* project engages several artistic practices and contexts in which I have worked over the years: (1) computational concrete and visual poetry; (2) retrocomputing, as it is somewhat problematically but frequently called, particularly on the Commodore 64; (3) sizecoding, the practice of writing very small computer programs to accomplish tasks that are thought to require much more code; (4) the demoscene, an international community of digital art creators originating in software cracking and piracy in Northern Europe in the 1980s; and (5) platform studies and practices. I will address each individually to explain the way I approached *256B for the C64 in 8K.* Then, I will mention some technical details and how this project connects to the art show theme.

1. Computational Concrete Poetry

Concrete and visual poetry relies on the reader’s visual encounter with the page to have its full effect; hearing a poem of this sort read aloud, if such reading is sensible, will not be adequate to the work. In the particular case of concrete poetry (as distinct from broader visual poetry practices, including shape and pattern poetry), an international movement beginning from multiple starting points in the 1950s, the poem is seen as not a picture but a structure, not words but a single word. What is seen on the page doesn't illustrate anything, but corresponds to the lines of force of the poem.[12] *256B* extends these ideas of concrete poetry with computation. My previous work related to concrete poetry is for a variety of platforms and in several programming languages, including Commodore 64 assembly. Beyond works in C64 assembly and BASIC, my programs/poems of this sort include the Concrete Perl series from 2011 [4,9]; “Use of Dust” from 2015 [4], based on the classic program/ poem “A House of Dust” [3]; “Una página de Babel” from 2015 [4], a component of the 2017 installation *Autofolio Babel;* and *Autopia* from 2016, which is a Web page and Python program [4] as well as a print-on-demand book [10] and gallery installation.

2. Retrocomputing

One does not really need to be “retrograde,” or look backwards, to use a computer such as the Commodore 64, which was manufactured and sold beginning in 1982. Like many media technologies, including letterpress printing, chemical photography, and audio recording on magnetic tape, “classic” computers are living technologies. They simply live in small and specialized niches, as with these other technologies. Letterpress printers produce wedding invitations rather than daily newspapers; artists and enthusiasts use instant chemical film from Fuji and Polaroid (previously the Impossible Project); sound artists and participants in cassette culture use magnetic tape, which is still manufactured, for audio; and demosceners and self-styled retro game developers still write new programs for the Commodore 64 and other computers of that era.

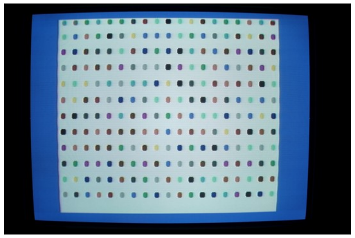


Figure 2: *After Damien Hirst* (2014), a one-line C64 BASIC program that was part of Programs at an Exhibition.

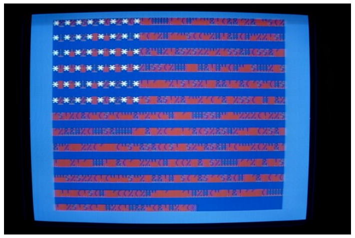


Figure 3: *After Jasper Johns* (2014), also a one-liner and in Programs at an Exhibition; shown at Boston City Hall as well.

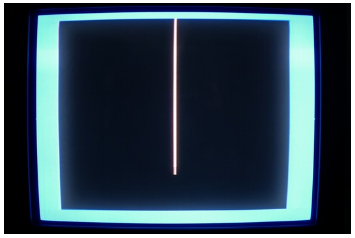


Figure 4: *After Barnett Newman* (2014), another one-liner. All of these use only the C64 character set, PETSCII, for visuals.

My involvement with home computers includes writing ad hoc programs on a variety of platforms that I keep set up in my home and in my lab/studio at MIT, The Trope Tank. I have written one 6502 assembly program, *Nanowatt* [8], responding to Samuel Beckett’s second novel, *Watt,* for the Commodore VIC-20. I have developed many artworks in assembly and BASIC for the Commodore 64. I featured five of these in a show at the Boston Cyberarts Gallery, Programs at an Exhibition (2014), where five Commodore 64s running one-line BASIC programs were displayed along with five running Perl programs by Páll Thayer. I also livecode visuals with musicians as part of LiveCode.NYC. I have done this at venues including PS1 and the Brooklyn club Sunnyvale, writing code from scratch on stage in BASIC, using two Commodore 64s and a video switch.

3. Sizecoding

Constrained composition has been a mainstay of my poetic practice, even when I am not writing computational poems. My first book of poetry, *Riddle & Bind,*[7] consists of literary riddles (the “Riddle” part), constrained poems (the “Bind” part), and some poems (the “&” part) that are constrained riddles. Size limitation is a simple but effective type of constraint in both standard writing and in coding. I have explored it in scholarship [5] and employed it often. Among my size-limited computational artworks are many demoscene productions and numerous one-line BASIC programs that are dedicated to poets—in addition to my series based on visual art. I have systematically explored sizecoding and creative computing in several ways. I led an effort with Trope Tank collaborators to regularly write, share, and discuss 256-byte creative programs over the course of a semester, which culminated in our hosting Salon 256, an event where people from the MIT community came to share and speak about such ultra-short programs.[1] I recnetl yfounded a literary magazine, *Taper,*[11] with an editorial collective. Two issues are out, the first including only HTML/JS/CSS up to 1KB in size; the second allowing work of up to 2KB.

4. The Demoscene

A context that involves “retrocomputing” (in the form of old school demos) and sizecoding is the demoscene, a computer art community seeking to impress a very technically aware audience. While most demoscene productions are not related to poetry, I have developed a demoscene practice (as nom de nom) that is, participating in parties in North America and entering a sizecoded (4KB) Commodore 64 production, done with a music collaborator, in the largest present-day demoparty, Revision, in Saarbrücken, Germany.[2]

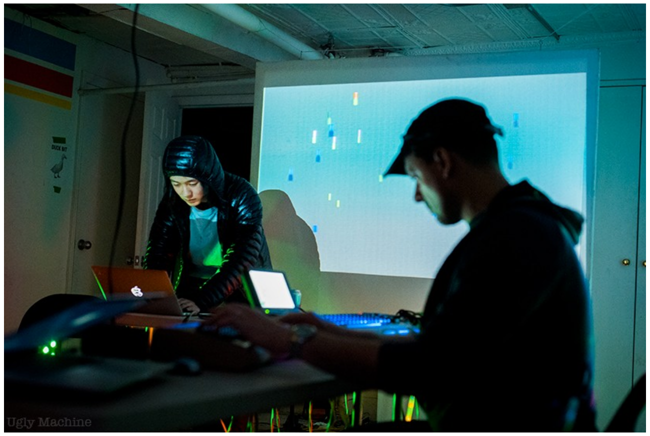


Figure 5: Sean Lee (left) and Nick Montfort at Babycastles during the opening concert of Synchrony 2018 on January 19. Lee is livecoding music using his platform, Gallium, while Montfort writes BASIC code live on two Commodore 64s to produce visuals on the single screen.

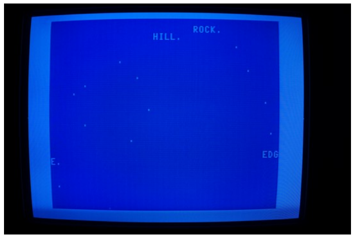
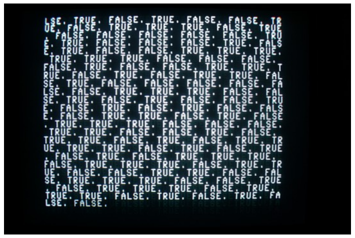


Figure 6: *Islets for J.R. Carpeter* (2015), a one-line C64 BASIC program responding to her ...*and by islands I mean Paragraphs* (2013).



```
LSE. TRUE. FALSE. TRUE. FALSE. FALSE. TR
UE. FALSE. TRUE. TRUE. TRUE. FALSE. TRUE
. FALSE. FALSE. FALSE. FALSE. FALSE. TRU
E. TRUE. FALSE. FALSE. FALSE. TRUE. FALS
E. TRUE. FALSE. FALSE. TRUE. TRUE. TRUE.
 TRUE. TRUE. TRUE. FALSE. FALSE. FALSE.
FALSE. TRUE. FALSE. FALSE. TRUE. TRUE. T
RUE. FALSE. TRUE. FALSE. TRUE. TRUE. FAL
SE. TRUE. FALSE. FALSE. FALSE. FALSE. FA
LSE. FALSE. TRUE. TRUE. TRUE. FALSE. FAL
SE. TRUE. FALSE. TRUE. FALSE. FALSE. TRU
E. FALSE. TRUE. FALSE. TRUE. FALSE. FALS
E. FALSE. TRUE. TRUE. FALSE. TRUE. FALSE
. TRUE. TRUE. TRUE. FALSE. FALSE. FALSE.
 TRUE. TRUE. FALSE. FALSE. TRUE. FALSE.
TRUE. TRUE. FALSE. TRUE. FALSE. TRUE. TR
UE. TRUE. TRUE. FALSE. TRUE. FALSE. TRUE
. FALSE. TRUE. FALSE. TRUE. TRUE. TRUE.
FALSE. TRUE. TRUE. TRUE. FALSE. TRUE. TR
UE. FALSE. FALSE. TRUE. TRUE. FALSE. FAL
SE. TRUE. TRUE. TRUE. FALSE. FALSE. TRUE
. FALSE. TRUE. TRUE. TRUE. FALSE. TRUE.
TRUE. TRUE. FALSE. TRUE. FALSE. TRUE. FA
LSE. FALSE.
```

Figure 7: *Answers to Legion for Craig Dworkin* (2015), a one-liner responding to conceptual writing.

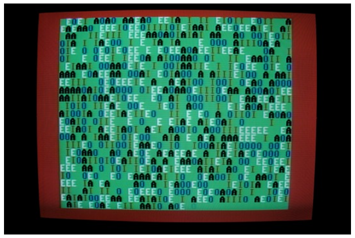


Figure 8: *Voyelles for Christian Bök* (2015), a one-liner exhibited at the Granoff Center.

To look beyond the Commodore 64, with a collaborator who focused on hardware and drivers, I developed a production, *Waves 3 Ways (Topsy’s Revenge)* that engages concrete poetry, sound/music, and also drives a Tesla coil. That one is among more than two dozen demoscene productions, including sizecoded JavaScript and Commodore 64 productions, that I worked on.[2]

I am also the founder and lead organizer of New York City’s first demoparty, Synchrony, which, the past three years, has involved starting in New York City, continuing on an Amtrak train over the course of a day, and concluding in Montreal. The event itself, in which a sundry group of digital media artists go from one country to another without being part of an official “group,” as far as Amtrak is concerned, is a border intervention as well as a demoscene event.

5. Platform Studies and Practices

Along with Ian Bogost, I developed the “platform studies” concept and am the founder and editor of the corresponding MIT Press book series. Bogost and I wrote the first book in the series, on the Atari VCS, a.k.a. Atari 2600.[6] This is a platform for which I have also written a 128-byte demoscene production.[2] The core idea of platform studies is that scholars should take the construct of the computer platform—whether it is a videogame console, a computer, or a software platform such as Flash—as seriously as they do particular games, national cultures and contexts, and historical periods. The platform’s design and development is intentional in response to social and cultural influences; the platform also influences society and culture, most directly by supporting and constraining creative practice on that platform. This direction has informed my work as an artist as well as to my scholarship, including my work on the Commodore 64.

Technical Aspects

It is important to the concept and spirit of this piece that a hardware Commodore 64 is used to execute the program. This is the way demoscene productions are almost always “screened” in the compos (competitions) near the end of a demoparty. It makes it clear that (for better or worse) *256B on the C64 for 8K* and works like it are highly specific software artifacts, not general and portable videos.

A Commodore 64 can produce RF, composite, and S-video outputs. The presentation of this piece will use composite (AV) NTSC output. It will use a standard composite-to-HDMI 720p/1080p upscaler so that output to the 8K projector can be provided. The video quality will, of course, have some rough aspects, but these will have to do with the materiality of the Commodore 64 and its VIC-II video chip, used to produce NTSC output, and the upscaling.

This piece (unlike some of my others, developed for different contexts of presentation) adheres to guidelines for flash thresholds such as the W3C’s [13] to avoid effects that can pose health hazards.

Conclusion

Artistic, literary, and computing practices that are often distinct have certain of their boundaries blurred and borders transgressed in my work. I also work to make a connection between my scholarly work on platform studies and the way I engage with platforms as a literary artist. *256B for the C64 on 8K,* with a very tiny program at its core, takes up a concern for interlacing in a few ways—not only dealing with the interlaced NTSC video signal, but also the weaving together of

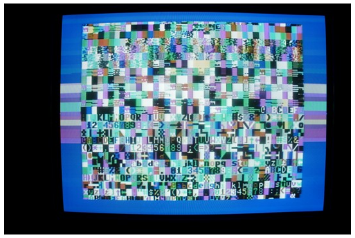


Figure 9: *Chronon* (2015), a 32-byte program written in assembly for the C64. This program is deterministic and uses data in ROM to provide glitchy/random-like visuals.

1980s “home computing” with late 2010s video technology, with the international cross-border connections made in the demoscene, and with the intersection of five different types of practice and context.

References

[1] Audry, Sofian, Angela Chang, Chris Kerich, Milton Läufer and Nick Montfort. 2017. “256-Byte Creative Programs.” Trope Tank Tech. Report TROPE-17-02, https://nickm.com/trope_tank/TROPE-17-02.pdf

[2] Demozoo. 2010–2019. [Entry in a demoscene site/ database.] “Nom De Nom.” https://demozoo.org/sceners/43383/

[3] Knowles, Alison and James Tenney. 1967. “A House of Dust.” [FORTRAN program.] Reimplementation by Nick Montfort, *Memory Slam,* November 14, 2014, https://nickm.com/memslam/a_house_of_dust.html

[4] Montfort, Nick. 2000—. *nickm.com.* [Personal and artistic website.] https://nickm.com

[5] Montfort, Nick. 2008. “Obfuscated Code.” *Software Studies,* pp. 193–199. MIT Press, Cambridge, Ma.

[6] Montfort, Nick and Ian Bogost. 2009. *Racing the Beam: The Atari Video Computer System.* Platform Studies series, MIT Press, Cambridge, MA.

[7] Montfort, Nick. 2010. *Riddle & Bind.* Spineless Books, Urbana, IL.

[8] Montfort, Nick. 2013. “Video of *Nanowatt* Online.” *Post Position.* https://nickm.com/post/2013/12/video-of-nanowatt-online/

[9] Montfort, Nick. 2014. *#!.* Counterpath, Denver.

[10] Montfort, Nick. 2016. *Autopia.* Troll Thread.

[11] *Taper.* 2018–. [Online literary magazine.] http://taper.badquar.to/

[12] Williams, Emmett, ed. and intro. *An Anthology of Concrete Poetry.* Something Else Press, 1967.

[13] W3C. June 5, 2018. “General Flash and Red Flash Thresholds.” Web Content Accessibility Guidelines (WCAG) 2.1. https://www.w3.org/TR/WCAG21/#dfn-general-flash-and-red-flash-thresholds