JESPER JUUL

TOO MUCH FUN

TOO MUCH FUN

TOO MUCH FUN

TOO MUCH FUN

TOO MUCH FUN

TOO MUCH FUN

TOO MUCH FUN

TOO MUCH FUN

TOO MUCH FUN

TOO MUCH FUN

THE FIVE LIVES OF THE COMMODORE 64 COMPUTER

Too Much Fun

Platform Studies

Nick Montfort and Ian Bogost, editors

Racing the Beam: The Atari Video Computer System, Nick Montfort and Ian Bogost, 2009

Codename Revolution: The Nintendo Wii Platform, Steven E. Jones and George K. Thiruvathukal, 2012

The Future Was Here: The Commodore Amiga, Jimmy Maher, 2012

Flash: Building the Interactive Web, Anastasia Salter and John Murray, 2014

I AM ERROR: The Nintendo Family Computer / Entertainment System Platform, Nathan Altice, 2015

Peripheral Vision: Bell Labs, the S-C 4020 and the Origins of Computer Art, Zabet Patterson, 2015

Now the Chips Are Down: The BBC Micro, Alison Gazzard, 2016

Minitel: Welcome to the Internet, Julien Mailland and Kevin Driscoll, 2017

Super Power, Spoony Bards, and Silverware: The Super Nintendo Entertainment System, Dominic Arsenault, 2017

The Media Snatcher: PC/CORE/TURBO/ENGINE/GRAFX/16/CDROM2/SUPER/DUO/ARCADE /RX, Carl Therrien, 2019

Who Are You? Nintendo's Game Boy Advance System, Alex Custodio, 2020

Four Shades of Gray: The Amazon Kindle Platform, Simon Peter Rowberry, 2022

Enacting Platforms: Feminist Technoscience and the Unreal Engine, James W. Malazita, 2024

Seeing Red: Nintendo's Virtual Boy, José P. Zagal and Benj Edwards, 2024

Intellivision: How a Videogame System Battled Atari and Almost Bankrupted Barbie, Tom Boellstorff and Braxton Soderman, 2024

Too Much Fun: The Five Lives of the Commodore 64 Computer, Jesper Juul, 2024

Too Much Fun

The Five Lives of the Commodore 64 Computer

Jesper Juul

The MIT Press Cambridge, Massachusetts London, England

The MIT Press would like to thank the anonymous peer reviewers who provided comments on drafts of this book. The generous work of academic experts is essential for establishing the authority and quality of our publications. We acknowledge with gratitude the contributions of these otherwise uncredited readers.

This book was set in Filosofia OT by New Best-set Typesetters Ltd.

Library of Congress Cataloging-in-Publication Data

Names: Juul, Jesper, 1970– author.
Title: Too much fun : the five lives of the Commodore 64 computer / Jesper Juul.
Description: Cambridge, Massachusetts : The MIT Press, [2024] | Series: Platform studies | Includes bibliographical references and index.
Identifiers: LCCN 2023057593 (print) | LCCN 2023057594 (ebook) | ISBN 9780262549516 (paperback) | ISBN 9780262380560 (epub) | ISBN 9780262380577 (pdf)
Subjects: LCSH: Commodore 64 (Computer)
Classification: LCC QA76.8.C64 J88 2024 (print) | LCC QA76.8.C64 (ebook) | DDC 004.165—dc23/eng/20240609
LC record available at https://lccn.loc.gov/2023057593
LC ebook record available at https://lccn.loc.gov/2023057594

10 9 8 7 6 5 4 3 2 1

d_r0

For NDB, universal accomplice.

Contents

Series Foreword

How can someone create a breakthrough game for a mobile phone or a compelling work of art for an immersive 3D environment without understanding that the mobile phone and the 3D environment are different sorts of computing platforms? The best artists, writers, programmers, and designers are well aware of how certain platforms facilitate certain types of computational expression and innovation. Likewise, computer science and engineering have long considered how underlying computing systems can be analyzed and improved. As important as scientific and engineering approaches are, and as significant as work by creative artists has been, there is also much to be learned from the sustained, intensive, humanistic study of digital media. We believe it is time for humanists to seriously consider the lowest level of computing systems and their relationship to culture and creativity.

The Platform Studies series has been established to promote the investigation of underlying computing systems and of how they enable, constrain, shape, and support the creative work that is done on them. The series investigates the foundations of digital media—the computing systems, both hardware and software, that developers and users depend upon for artistic, literary, and gaming development. Books in the series will certainly vary in their approaches, but they will all share certain features:

- a focus on a single platform or a closely related family of platforms
- technical rigor and in-depth investigation of how computing technologies work
- an awareness of and a discussion of how computing platforms exist in a context of culture and society, being developed on the basis of cultural concepts and then contributing to culture in a variety of ways—for instance, by affecting how people perceive computing.

Preface

The Commodore 64 computer came in a friendly box with pictures of people using it. My parents brought it home in 1983 after months of persistent entreaties (“I want a home computer”) had worn them down. We connected the machine to the family TV and took turns typing commands in the BASIC programming language. Tape drives were out of stock, so we had no way to save our programs, and I wrote mine down in a little notebook.

But what did I think this machine was? And what was it for?

As a matter of fact, I already had some experience with computers outside of the home. Prior to 1983, my parents had taken me to Aarhus University where I had tried programming on computer terminals with glowing green letters. Computers in the home were rare, and I was filled with thirst for a computer of my own, to program and experiment.

Video games also spurred my imagination. There were games in the arcade—*Space Invaders*, *Galaxian*, *Frogger*, *Gorf*, *Pac-Man*—years ahead of games in the home, and video game consoles like the Atari 2600 were rare in northern Europe. Home computers promised a path to video games, freeing me from spending precious coins for every try, from visiting seedy cinema lobbies and dubious fast-food shops.

And so, with the arrival of the C64 in our family home in 1983, I could turn myself actively towards programming and video games; programming video games. The C64 taught me how to program, yes, but I admit to playing countless games on it. It taught me about them too.

This book came about when I realized I had forgotten something I used to know, about early computer and game history, about the strange world of home computer games, bedroom coding, and wild experiments that we, and I, had been neglecting when painting big picture computer and video game histories. With this book I aim to recover some of the history that we forgot.

I am indebted to my research assistants, Laurel Carney at MIT (C64 advertising study and writeup) and Lucas Johannes Fellner and Jonathan Skjøtt at the Royal Danish Academy (*Zzap!64* study).

Thank you to the readers of parts, wholes, and iterations of the manuscript: Graeme Kirkpatrick, Emil Lundedal Hammar, Olli Sotamaa, Mia Consalvo, Kim Forum Jacobsen, Nick Montfort, Nanna Debois Buhl, Susana Tosca, Chaim Gingold, Gleb J. Albert, and Alessandro Canossa.

Thanks to Jesper Lagerberg, Michael Debus, Noah Falstein, Arthur Steijn, Markku Reunanen, Akito Inoue, and Matteo Bittanti for discussions.

Thanks to Kasper Ernborg Gerlach for illustration design.

Thanks to MIT Press editor Noah J. Springer, series editors Nick Montfort and Ian Bogost, Elizabeth Agresta, and the rest of the MIT Press for trust and belief in the project and to the MIT Press reviewers for insightful and useful comments.

Thanks to my colleagues at the Institute for Visual Design at the Royal Danish Academy in Copenhagen for running commentary and encouragement.

Elements of this book were first presented at the New York University Game Center, The Trope Tank at MIT, the History of Games Conference 2022, the 2023 Tampere University Game Research Lab Spring Seminar, and the DiGRA 2023 conference in Seville.

The outline of the book was written during my research stay in spring 2022 at the Massachusetts Institute of Technology Comparative Media Studies/Writing Trope Tank, hosted by Nick Montfort. The stay at MIT was supported by the Carlsberg Foundation. The publication of the book was supported by the Carlsberg Foundation, grant CF23–0570.

Non-English sources were translated by the author.

The book's accompanying website features supplemental material, including videos, source code, and C64 programs running in emulation.

jesperjuul.net/c64/

The ► character signals that an example can be found on the website.

I have tried to keep technical jargon to the necessary minimum. Should any term be unfamiliar, the end of the book contains a glossary of technical terms, central people, and central computers.

Introduction: The Five Lives of the Commodore 64

There is an immediate friendliness to the dark- and light-blue interface that meets you when turning on a Commodore 64 (C64) (figure 0.1), to the BASIC programming language, designed to be simple and accessible, and to the special color and sound abilities of the machine (figure 0.2). Once you have memorized a few commands, the C64 is an infinite playground for experiments like the universal trick of printing your own name in colors across the screen. (► Write a program printing your name on the book's website.)

I want to tell the story of a computer that changed, though it stayed the same. The Commodore 64 was not functionally updated in the twelve years it was produced, from 1982 to 1994. It did not become faster or gain more memory, but the C64 was considered to be many different things, and figure 0.2 shows a screenshot from each of the five phases of the machine's history, *The Five Lives of the Commodore 64*, that I am writing about:

- **First life**: A user buying the C64 in 1982 might emphasize its ready-for-business, forty-column screen and its ample 64 KB of RAM and use it for BASIC programming and word processing.
- **Second life**: In the mid-1980s, another user could be thinking of the C64 primarily as a computer for games like *International Karate+*, unlike the boring PCs of the time.
- **Third life**: In 1987, it could be a computer for access to a culture, the *demoscene*, of people using the C64 for impressive new graphical programming.
- **Fourth life**: The same year it could also be an aging computer with a new graphical interface software upgrade that let it compete (somewhat) with the new Macintoshes, Amigas, and Atari STs.
- **Fifth life**: As I write this in early 2024, the C64 is a comforting device from the past whose simplicity makes it ripe for hobby projects and

whose visuals and sound, while not cutting-edge, do not feel dated in the way a merely ten-year-old computer might. The C64 now appears deliberate, like a *style*, like a set of design choices with a distinct personality.

Figure 0.1 1980s Commodore 64 setup with printer, disk drive, monitor, joystick, and modem. (Displayed at the Vintage Computer Federation Museum in New Jersey; photo by the author.)

Already in 1983, Ken Williams of game developer Sierra Online was spreading the rumor that Commodore was planning to discontinue the C64 after the holiday season, but the machine was produced for another eleven years.[1] The Commodore 64 had more lives and lived much longer than anyone imagined. This leads to the core question of the book: *Given that the Commodore 64 did not change, how could it be so many things to so many different people, for so long?*

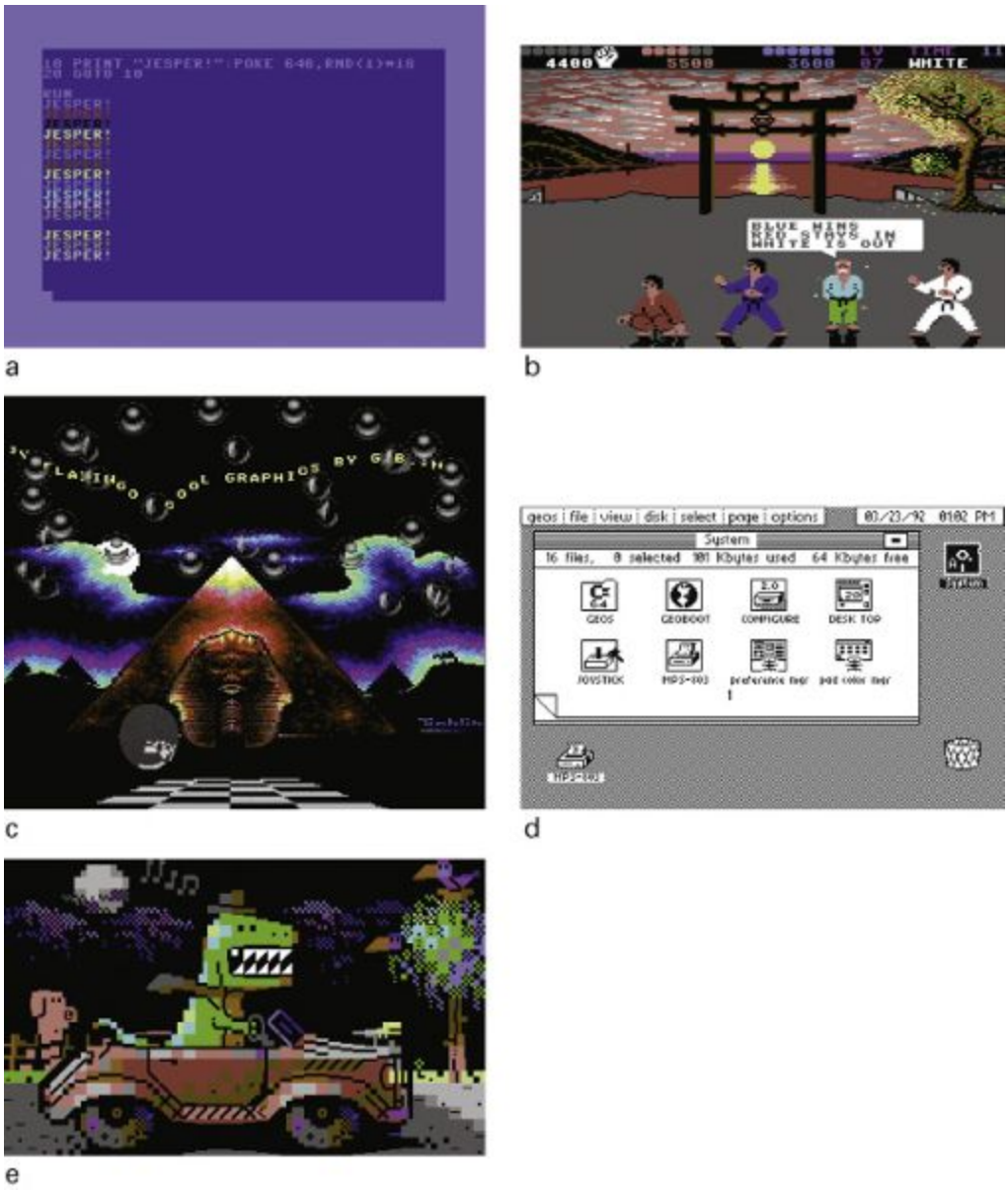


Figure 0.2 The Commodore 64 machine was not upgraded or made faster since launch, but over time it was imagined to be five different things: a) A computer for productivity and programming in the BASIC language, often starting with printing the program author's name; b) A machine for games, here *International Karate+* (System 3, 1987); c) A machine for creating "demos" with "impossible" technical tricks, here *Brutality* (Light, 1991); d) An aging machine that could keep up with newer computers through the GEOS graphical interface (Berkeley Softworks, 1986); e) A machine whose limitations are now a historical *style* that artists can revisit—for example, *Gary* (Genesis Project, 2013).

The C64 was arguably the most powerful home computer for video games of its time. It is famous for its video games (figure 0.3) like *Boulder Dash*, *Sid Meier's Pirates!*, *Monty on the Run*, *Paradroid*, *SimCity*, and *Bubble Bobble*. In the 1980s, though, it was commonly believed that the games of the Commodore 64 computer were a problem. The dismissive label of "game machine" implied that because the C64 was good for playing games, it was "too much fun" to take seriously and could not also be used for "serious" things like word processing (which it *was* used for). The label also meant that if you wanted to be "serious," you should get a much more expensive computer with nearly identical capabilities except for the ability to play good games. The stigma of the "game machine" was a constant worry for users and manufacturers.

In my professional life, I am mostly a video game researcher, but the C64 played a significant role in my life, and to go back to it is to revisit my role as a fan then.[2] I spent formative years with the C64, the typical setup of which is shown in figure 0.1. I have traversed its depths, exploited the quirks of it chips, played games that others made, and made games myself. As a teenager, I assumed the now-awkward handle, or pseudonym, The Cracking Dane (TCD), and felt the joy of working late on the illegal practice of removing the copy protection from—"cracking"—a game. I created audiovisual programs in machine language—demos—to show off my skills. With friends I participated in the demoscene, going to "parties" where we would show our latest creations and sleep on the hard floor of school auditoriums.

I have included my own experiences to provide first-person context to the C64's history. While being an insider in C64 culture has advantages such as access and an understanding of the machine and the jargon, Finnish demoscene researcher Markku Reunanen has noted how it can also lead to blind spots such as dislikes and overconfidence in the breadth of one's knowledge.[3] Like Reunanen suggests, I have tried to select examples not via my personal tastes, but through interviews, magazines, and archival material. The brunt of the material here is from other people and sources, but I use my own experience to illustrate the larger history.

The Commodore 64 is officially the best-selling desktop computer model of all time according to the Guinness Book of World Records, with most likely 12.5 million units sold, most in Europe, followed by the US,

Australia, and New Zealand.[4] Figure 0.4 shows the sales of the most popular home computer platforms from 1977–1994, with the C64 being by far the most popular, followed by three platforms having sold around 5 million units—the ZX Spectrum, the Apple II, and the Commodore Amiga, and three platforms around 4 million—Atari 8-bit, the C64-compatible Commodore 128 (C128), and MSX.[5] Some readers, especially from the US, may be surprised to learn this, as they are used to hearing about the centrality of the Apple II in computer history. Much higher C64 sales figures have been reported, but a double study of Commodore financial reports and C64 serial numbers point to the same conservative figure of 12.5 million.[6] According to the MobyGames database, the C64 is also the platform for which most games were published between 1977 and 1990 (figure 0.5). Furthermore, in 1985 the C64 surpassed the Apple II as the platform with the largest accumulated game catalog and retained that position until overtaken by MS-DOS in 1994 (also figure 0.5).

Figure 0.3 The C64 is fondly remembered for its games: a) *Boulder Dash* (First Star Software, 1984); b) *Sid Meier's Pirates!* (MicroProse, 1987); c) *Monty on the Run* (Gremlin Graphics, 1985); d) *Paradroid* (Hewson, 1985); e) *SimCity* (Wright, 1989) and f) *Bubble Bobble* (Software Creations, 1987).

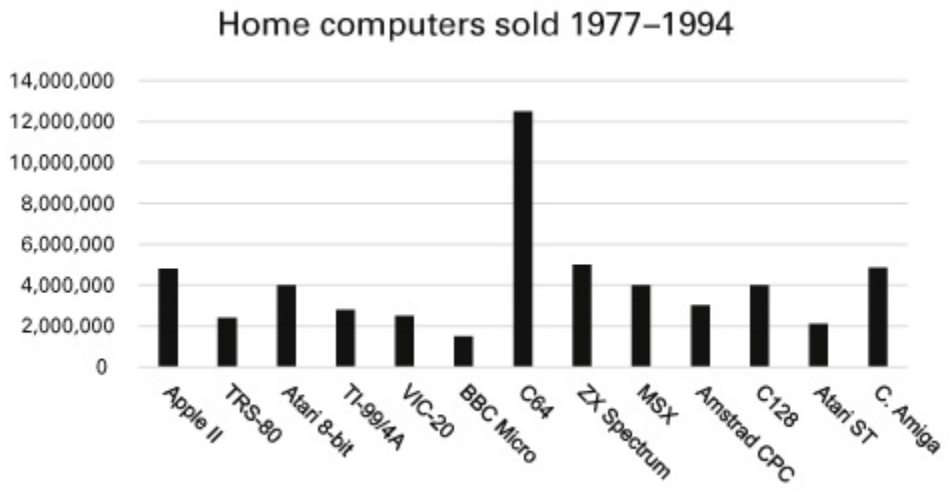


Figure 0.4 Home computers sold worldwide (from 1977–1994), showing the C64 as the by-far best-selling home computer of the era.

Home computer: Mass-market computers in the 1980s were usually referred to as "home computers" as opposed to business computers.

Microcomputer: In the UK, the term "microcomputer," or "micro," was more common, but was largely synonymous with home computer.

Personal computer: Broader term that encompasses both home and business use but is also used to imply a computer not used for games.

Business computer: Computer designed solely for business purposes.

Reading most video game histories, you would not know that the C64, or home computers in general, played any significant role in video game history. Steven Kent's *The Ultimate History of Video Games* claims to cover all digital games but includes no computers in its timeline and gives home computers a miniscule fraction of the coverage of Nintendo, Atari, or Sega.[7] Kent's history is characteristic of the US version of history that centers on the "video game crash" of 1983–1985 which devastated the US video game console industry and led to the subsequent entry of Nintendo into the US home console market. Tristan Donovan's *Replay* attempts to go beyond that standard history, but still gives five times the space to each Nintendo and Atari than to Apple, Commodore, or the ZX Spectrum.[8] There is a staggering discrepancy between the number of games published per

platform and the games covered in such histories. Video game history is often reduced to a few successful companies—Nintendo, Sony, Microsoft, Atari—and home computers all but disappear.[9]

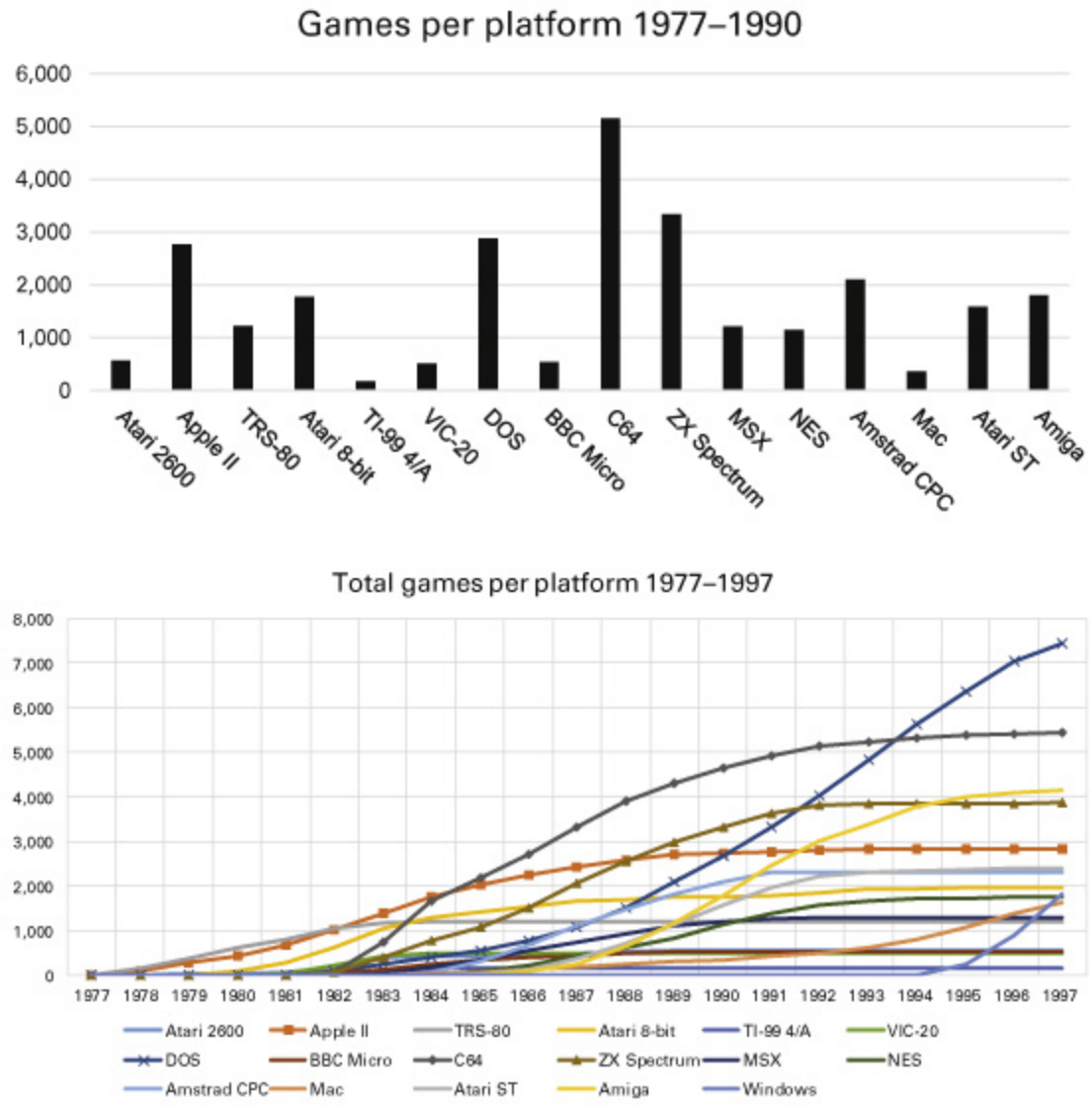


Figure 0.5 Games published per platform, showing the Commodore 64's predominance among its contemporary home computers and video game consoles; accumulated games per platform 1977–1997, showing the C64 overtaking the Apple II in 1985 and being overtaken by DOS in 1994.

Computer historian Thomas Haigh notes that popular books on computing are centered on "people and stories that readers already know about while nevertheless insisting they are tragically forgotten."[10] The C64

is not forgotten, but it is also strangely missing from some *computer* histories, and Commodore is, according to Haigh, "largely ignored by historians and journalists alike because its breakthrough hit was an inexpensive machine for home use."[11] In a 1994 obituary for the Commodore company, journalist Thomas Halfhill lamented that Commodore made pioneering devices but was forgotten by "revisionist historians who claim that everything started with Apple or IBM."[12] As Haigh points out, even Paul Ceruzzi's excellent *A History of Modern Computing* discusses home computers and repeats well-known stories of Apple and RadioShack, but mysteriously fails to mention the C64.[13] Thus, the present book is also a detective story about how the C64 went missing in history, and in it, I aim to recover some of the history of both video games and computers.

Having grown up with the C64, I have always felt a disconnect from such mainstream video game and computer histories on a personal level—the C64 was the dominant platform of northern Europe in the mid-1980s, and IBM, Apple, Atari, and Nintendo barely registered. There is even local European pride in the C64, as US as its origins may be: the Technical Museum in Berlin displays a 1986 gold C64 commemorating the sale of 1 million C64s in Germany (figure 0.6). In Finland, the Commodore 64 was known as *tasavallan tietokone*, "the computer of the republic."[14]

The C64's story is thus quite different depending on where you were in the world. As will be shown in chapter 2, the Commodore 64 was grouped with, and competed against, many other computers, all shown in figure 0.7.[15] In the US, the main competitors were the Apple II, TRS-80, and Atari 8-bit computers. In the UK, the Commodore 64 was up against the BBC Micro and the ZX Spectrum, and the C64–Spectrum rivalry was sometimes known as "The Great British Computer War," though it also took place across much of Europe, Australia, and New Zealand.[16] In most of western Europe, the C64 was dominant but competed against the ZX Spectrum and later the Amstrad CPC 464, though less so in France (Thomson computers) and Spain (MSX). Hungary had a late period of C64 development in the early 1990s. In Japan, the Commodore VIC-20 had sold well, but the Commodore 64 was marginal compared to MSX, PC-8800, and other computers. These facts determine the geographical focus of this book:

North America, northern Europe, Germany, the United Kingdom, Italy, Australia, New Zealand, and a few other countries.

Figure 0.6 The Commodore 64 was especially popular in Germany. The German Museum of Technology in Berlin exhibits a 1986 gold C64 celebrating the 1 millionth machine sold in (West) Germany. (Photo by the author.)

Perhaps, it was just that the C64 was much more popular in Europe, explaining why US histories forgot about it? When researching for this book, however, it became clear that the US market of the mid-1980s was itself dominated by Commodore machines (figure 0.8), with by far the largest home computer market share (31 percent) in 1986.[17] In their report on the first online world, *Habitat* for the C64, the developers explained that, "At the time we started (1985), the Commodore 64 was the mainstay of the recreational computing market."[18] North Americans appear to have forgotten their own history. Demonstrating not only sales but Commodore's cultural mindshare, the 1985 *MAD Magazine* spoof of "The Computer Company President of the Year"[19] featured not any now-famous computer figure like Steve Jobs or Bill Gates, but a caricature of Commodore's Jack Tramiel, now president of the "Commode" company. The spoof makes fun of Commodore's reputation for making "overpriced 'Game' machines."

There are three main culprits for the strange erasure of Commodore and other companies in history: First, Steven Levy's 1984 *Hackers: Heroes of the Computer Revolution*—which did not mention the C64, despite being published while it was the best-selling computer in the US—established the now-familiar story of hacking as a straight line starting at the Tech Model Railroad Club at MIT in the 1950s, moving through the Altair 8800 computer, through Microsoft developing Microsoft BASIC, through the Apple I and II, and to game developers for especially the Apple II and the Atari 2600.[20] Second, Robert X. Cringely's 1992 book *Accidental Empires*, later adapted as the 1996 PBS documentary *Triumph of the Nerds*, simplified computer history to a few heroic but weird individuals and ranked companies using idiosyncratic inclinations such as, "Apple took a heroic approach to selling the personal computer and, by doing so, taught all the others how it should be done."[21] IBM is given some credit, but all other companies are vaguely dismissed as not "changing the world," showing that this simplistic history requires much hand-waving to maintain.[22] Third, the 1999 television drama *Pirates of Silicon Valley* completed the trajectory and reduced computer history entirely to a rivalry between Bill Gates and Steve Jobs, only sometimes mentioning IBM.[23] Since then, the standard story has chugged along, and the timeline of Walter Isaacson's 2014 *The Innovators* includes as only packaged computers the Apple II, the IBM PC, and the Macintosh, adding some photos of Bill Gates.[24] If these are your main sources, you will be entirely surprised to hear of Commodore's 1986 US market share, that the C64 was the best-selling home computer, or that *MAD Magazine* chose to parody Commodore.[25]

The more general problem with such histories is that—to tell a good story—they make computer history to be about heroic individuals touched by flashes of insight. The subtitle of Apple engineer Steve Wozniak's autobiography *iWoz* is, *How I Invented the Personal Computer, Co-Founded Apple, and Had Fun Doing It*. In it he recounts a singular moment where he made the personal computer come to fruition, "I didn't realize it at the time, but that day, Sunday, June 29, 1975, was pivotal. It was the first time in history anyone had typed a character on a keyboard and seen it show up on their own computer's screen right in front of them."[26] If this story of invention for a moment sounds plausible, it is because we can

picture him working alone in a garage, and we are able to forget the people who designed and manufactured the parts from which his computer had been assembled. By comparison, Commodore designed the C64 chips in-house, and, as we will see, it is clear that the C64 was designed by many different people. It is also notable that Wozniak's story omits not only the components and ideas that went into his prototype, but also all the already-existing, if expensive, packaged personal computers like the 1970 Datapoint 2200 and the 1973 Xerox Alto (figure 0.7).[27]

Joy Lisi Rankin call the standard simplistic story "the Silicon Valley mythology," where all the rich history of computing, including the creativity on early time-sharing platforms, is erased in favor of a few geniuses inventing everything mostly in California garages.[28] It is not just Commodore that is forgotten—in his book about the creation of the IBM PC, Dennis Andrews similarly laments that "IBM's contribution has been forgotten to a large extent," and it is true that few people know how it was created.[29]

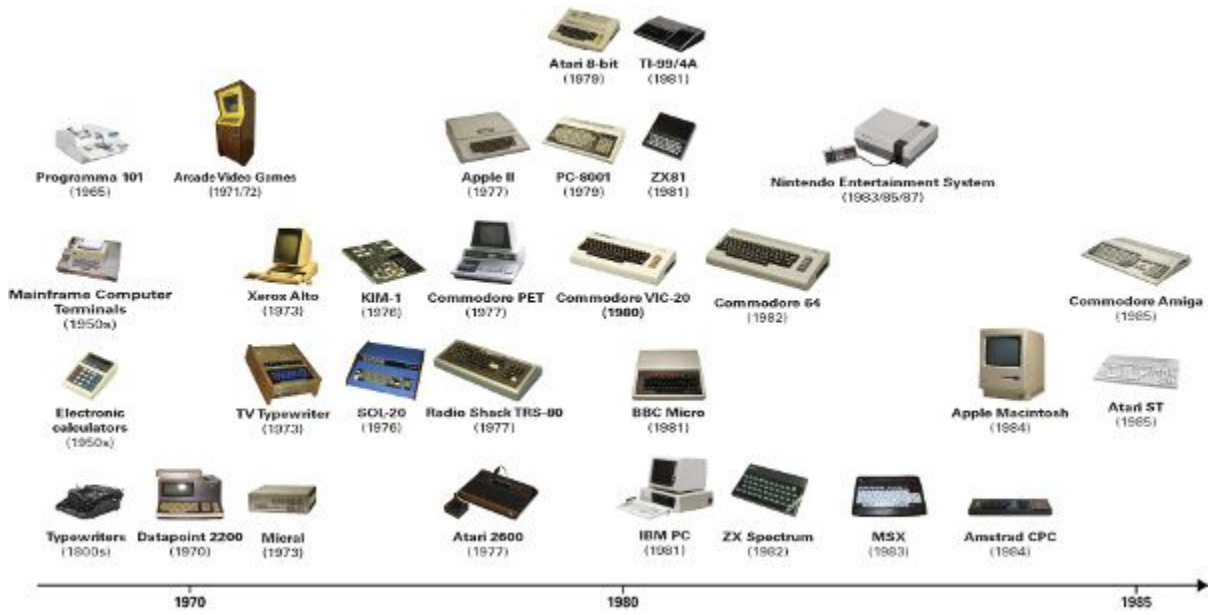


Figure 0.7 The Commodore 64 was built on ideas from previous computers, in turn built on ideas from mainframe computers, hobbyist engineering kits, calculators, typewriters, and video games. The C64 in turn influenced later computers to include fast color graphics and sound.

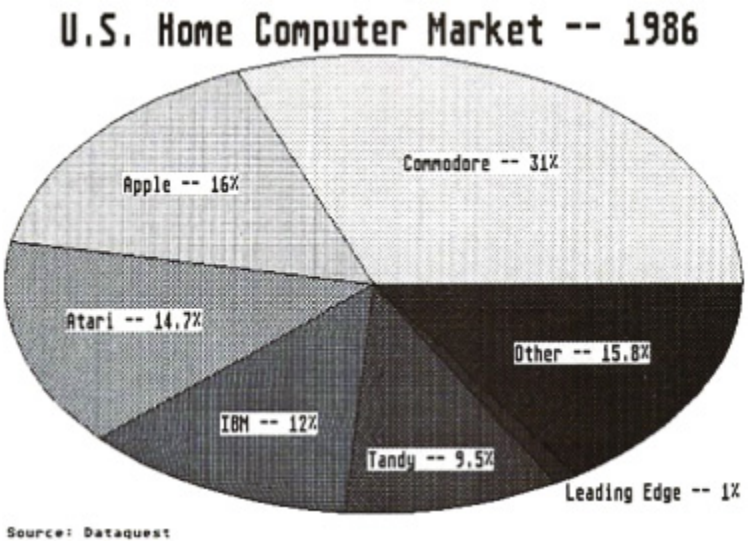


Figure 0.8 Dataquest survey of US home computer market share in 1986, showing Commodore as the most popular manufacturer.

The five-lives structure of this book also serves to make two points here: the general point that a given computer can play many different roles, and the specific point that the Commodore 64's central contribution of adding colors, sound, games, and *fun* to computers is also a core reason why it is often erased in computer history. Chapters 1 and 2 show how games were central to the attraction of the machine, but chapter 3 shows that programming also became an attractive and social activity through the demoscene, chapter 4 shows the effort directed towards keeping the games of an aging machine up to date, and chapter 5 shows the new enjoyment of returning to an old computer. Some writers of computer history have clearly seen themselves as above the trivialized matter of a cheap, popular, colorful computer and its games and culture, and I will try to make up for this omission.

Before the Commodore 64

Commodore was first incorporated in Toronto in 1958 as a typewriter company, later moving into calculators and finally into computers in the late 1970s. The best-known Commodore figure, CEO Jack Tramiel (figure 0.9), was an Auschwitz survivor who immigrated to the US in 1947 and became known as a businessperson with a talent for making deals. The

Commodore story is interesting in part because of how poorly it fits common preconceptions of what computer history should be. *Where* precisely Commodore fits is a question in itself. Commodore's main activities were not near Silicon Valley, nor in Massachusetts, home to many earlier computer companies. Instead, during its most successful years, the company's operations were headquartered in West Chester, Pennsylvania. However, the C64's predecessor, the VIC-20, was partially developed and first launched in Japan, and its case was the basis for the Commodore 64. The C64 and later Amiga were more popular in Europe than elsewhere. Thus, Commodore, which shuttered in 1994, is geographically unmoored.

Figure 0.9 At Commodore's 1980 meeting in London, the company was torn about its future direction. 6502 chip designer Chuck Peddle (right) wanted to make business computers, but CEO Jack Tramiel (left) decided to aim for the home market with the VIC-20. (Photo © Michael Tomczyk.)

How did a typewriter company become a computer manufacturer? It is easy to forget, but prior to 1977, computers were not produced for home users. Figure 0.7 shows how home computers were part of a longer and larger computer history, including mainframe computer terminals, circuit boards like Commodore's KIM-1, and microcomputers like the French Micral. Commodore's first computer, the PET, came to be because chip designer Chuck Peddle (figure 0.9) had unsuccessfully tried to convince his

employer MOS Technology to build a computer (MOS already sold the KIM-1 computer board).[30] After Commodore acquired MOS, Peddle suggested to his new boss Jack Tramiel that the company make a computer. Jack Tramiel was not well-versed in the topic and let his son Leonard Tramiel discuss the proposition with Peddle. According to the younger Tramiel, the conversation centered on science fiction, which was the impetus for making a computer in the first place:[31]

> The topic of the conversation was the Robert Heinlein short story *The Door into Summer*. He [Peddle] wanted to live in that world with ubiquitous computing. In order to be comfortable with ubiquitous computing, people need to be comfortable with computers and robots. To do that you needed a low-cost and easy-to-use computer. You needed a low-cost microprocessor. He had done the microprocessor. The computer was next.[32]

Science fiction thus played a central role in the conception of Commodore's first computer.[33] Its full name, Commodore PET 2001, had a clear reference to Stanley Kubrick's film *2001: A Space Odyssey* (1968), whose screenplay was cowritten with Arthur C. Clarke.[34]

1977 saw the release of the first three mass-market and preassembled computers, later named "the trinity" by *BYTE*: the RadioShack TRS-80, the Apple II, and the Commodore PET.[35] With its built-in black-and-white monitor and direct start of the BASIC interface, the PET mimicked office machines and the computer terminal, while the Apple II and TRS-80 were closer to the hobbyist electronics tradition in that they could be opened and modified and were designed for connection to a TV, and in that they required special keystrokes to launch BASIC. These packaged computers expanded the market for computers from engineering hobbyists to regular consumers.[36]

After the relative success of the Commodore PET, the company had to decide what to do next. At a famous April 1980 meeting of Commodore executives and top engineers at the Burnham Beeches mansion near London (figure 0.9), the Commodore company was torn about its future direction. Should Commodore continue the business- and education-oriented PET

series, develop a new high-end line of computers, or aim for the consumer market? CEO Jack Tramiel pounded the table and declared that the company would build a low-end computer for the home, to the dismay of many engineers and managers.[37] The resulting VIC-20 computer was released at the end of the year in Japan, and in the rest of the world the following year. The VIC-20 was colorful and cheap, connected directly to a TV rather than to a dedicated monitor, and was marketed as "The Friendly Computer," the ad campaign featuring William Shatner of *Star Trek* fame.[38] The VIC-20 outsold Commodore's PET line as well as its contemporaries the Apple II and the TRS-80 and became the first computer to sell a million units.[39] Its success proved that there was a market for low-cost color computers.

A surprising number of former employees have written about working for Commodore, notably US product manager Michael Tomczyk, UK director David Pleasance, US engineer Bil Herd, and German manager Petro Tyschtschenko.[40] Commodore's overall history, most thoroughly gathered in Brian Bagnall's trilogy *On the Edge: The Spectacular Rise and Fall of Commodore*, *Commodore: The Amiga Years*, and *Commodore: The Final Years*, gives the impression of a meandering company, whose centrally planned development projects are sometimes superseded by personal projects of single engineers, a company that fails to retain the talent behind their most successful computers.[41] Tramiel was said to be charismatic but volatile, and engineers have reported "Jack attacks," personal scoldings received whenever Tramiel was unhappy.[42] This even gave the name to Commodore's 1983 game *Jack Attack*. Prone to launching lawsuits and alienating partners, Tramiel was later ousted and eventually replaced as CEO by Irving Gould, known for spending much of his time in the Bahamas.[43] Tramiel is remembered as a business innovator, claiming that "business is war" and that the company was making "computers for the masses, not the classes" (one of Commodore's innovations was to sell computers in toy and department stores rather than just in computer stores).[44]

Commodore was a large company with considerable technological abilities that far surpassed competitors like RadioShack, Sinclair, Atari, Apple, or Amstrad. Having bought chip design and manufacturing company MOS Technology, Commodore was, alongside Texas Instruments, unique in

being a home computer manufacturer with its own chip development and fabrication capabilities. This gave Commodore the kind of vertical integration we today associate with Samsung or Apple, allowing it to develop and produce custom-designed computers with features not available elsewhere, and at lower cost than the competition. Chips developed by Commodore's MOS were among the most dominant of the era, with the 6502 family of CPUs running not only Commodore's own machines, but also the Atari 2600, Apple II, BBC Micro, Atari home computers, and the Nintendo Entertainment System.

The history of Commodore also includes momentous decision points that could have made history wholly different. In 1976, Jobs and Wozniak tried to sell their Apple II computer to Commodore for "a few hundred thousand dollars" only to be turned down.[45] Commodore also rejected the offer to distribute VisiCalc, the first spreadsheet.[46] It seemed logical that Commodore would license Nintendo's games, given Commodore's Japanese connections, and given that many early VIC-20 games had been developed by Japanese HAL Laboratory, including by later Nintendo boss Satoru Iwata.[47] However, in 1982 product manager Michael Tomczyk negotiated such a deal with Nintendo, only to have Tramiel renege on the agreement at the last minute with no explanation. Tomczyk has explained his frustration with Tramiel's decision, and attributes it to Tramiel not wanting to "insult or offend Bally," the then-criminal reputation of which Tomczyk confirms, "Some people claim that there were some tough characters running Bally/Midway, and I can verify that."[48] Nintendo made deals with Atari and Coleco instead. It is easy to second-guess such decisions, of course, but we cannot know how the alternative histories would have played out.

Imaginaries and Resistant Computers

In a 2000 short story collection covering his long career, science fiction author Arthur C. Clarke observed that, "roughly a third of these stories were written when most people believed talk of space flight was complete lunacy. By the time the last dozen were written, men had walked on the Moon."[49] Sci-fi authors like Clarke or Jules Verne wrote about space travel well before any space travel had taken place, but their stories influenced

engineers and decisionmakers and thus paved the way for actual space travel.[50] Peddle developed the Commodore PET in part to bring about an imagined future containing computers and robots. The field of media archeology calls such ideas *imaginaries* (noun) to describe how ideas and imagination play a key role for media and technology. Some imaginaries become actual devices, some remain an idea, and some are partially realized. Media archeologist Jussi Parikka argues that we should take interest both in what is actual and what is impossible, and think of technology history as more than a series of successful products.[51]

Following Eric Kluitenberg's concept of *imaginary media*, we can see that even actual inventions and actual media have imaginary components.[52] Voice-activated virtual assistants today were inspired by talking computers in cinema (like HAL in *2001*), but they nevertheless lack the intelligence and consistency that cinema has promised us. Virtual reality was developed in a culture which had seen the 1992 movie *The Lawnmower Man*.[53] Wearing a VR headset today, we may imagine ourselves part of a future where we spend most of our time in virtual spaces, but we then often remove the headset after ten minutes. Imaginaries often set up expectations that exceed what a machine can do, but the actual and the imaginary work closely together, and "the transition between imaginary and actual media machines . . . can be almost seamless."[54]

In this book, I take *imaginary* to mean *any imagined concept of a technology's function, role, or meaning*. Imaginaries also shape our interaction with a computer. In fact, computers give us two strong and vastly different experiences when we engage with them, and these experiences were especially strong when home computers were a new idea.

1. **Imagined computers**: The first experience is reading, hearing, or seeing ads about a new computer. You have not touched the device, but you are imagining what it can do, the world it will create, how it will fit into your life. You approach it in anticipation of this new world coming about.
2. **Resistant computers**: The second experience is to use a computer and find it resisting and guiding us. Consider the experience of planning a grand new computer program but becoming stuck due to the common

mistake of confusing the letter o for the number 0 when typing, yielding only an enigmatic error message.

```
X=42/10
?SYNTAX ERROR
```

In this way, the machine pushes us in directions of its own: We may plan to draw a large triangle on the C64 screen, but the minimal BASIC programming language and the layout of the computer's memory make it difficult for us. It is easy to change colors though, and easy to move sprites around the screen, and the computer seems to encourage us to use it in certain ways.

The two experiences, of imagining a computer and the computer resisting, can explain how it is possible for a physically unchanging machine to seemingly change anyway. First, researchers Simone Natale and Gabriele Balbi make the point that imaginaries and fantasies often change over time for a medium or technology—for example, radio has been imagined to be many different things since its invention.[55] Second, we can use philosopher Jane Bennett's *new materialist* ideas about the ability of things to act by themselves to observe that a computer can have an unintentional agency of its own. The C64 is not just its design, manufacturing, advertising, or our imaginary ideas about computers; it also acts on its own, even doing things not intended by its creators. Bennett calls this *vitality*, "the capacity of things . . . not only to impede or block the will and designs of humans but also to act as quasi agents or forces with trajectories, propensities, or tendencies of their own."[56]

This is the endless loop between the imaginary machine and the concrete machine: Over time we hear new things about the C64. When we use the machine again, we see the C64 in a new light and approach it in a new way. We experience new things as the microscopic features of a chip in the machine respond to our new behavior, and we change our imaginary again and communicate it to others, ad infinitum. It can sound like a paradox, but it really happened. Without physically changing, the C64 seemed to change, but as we will see, the physical C64 played a leading role in this evolution.

How the C64 Was Designed

Commodore’s three biggest successes, the VIC-20, C64, and Amiga, all began as video game–focused chip projects only to be turned into computers.

So, the C64 was *designed* for games? We can look inside the machine and see the concrete results of the ideas and intentions of its designers. Common histories of computing often tell us of individuals with singular visions, visions then made manifest in technological devices. But opening the C64 to examine the circuit board (figure 0.10), we realize that the machine is constructed from chips designed for several entirely different purposes. The two rows of chips on the bottom left are the 64 KB of RAM giving the machine its name (see the glossary for an explanation of technical terms). But the three most important chips here, the 6510 CPU, the 6581 SID sound chip, and the VIC-II graphics chip are all placed next to the open silvery box on top, with its connectors to TVs and monitors. As it turns out, each chip was designed with a specific imaginary in mind, and *none of these were about a home computer*. The 6510, part of the 6502 family, was designed for control systems, the 6581 SID was designed for music synthesizers, and the VIC-II graphics chip was originally designed for video game systems. How did all this lead to a computer?

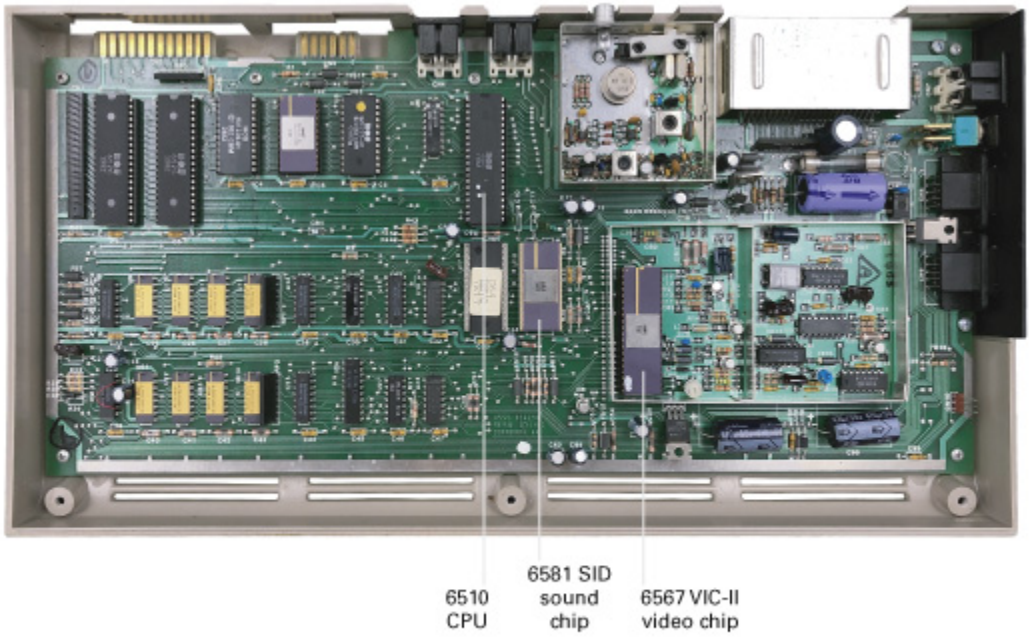


Figure 0.10 The circuit board inside an early C64. The 6510 CPU was designed for control systems, the 6567/69 VIC-II graphics chip for video game machines, and the 6581 SID sound chip for music synthesizers. None were originally designed for a computer. (Photo by David McMurtrie.)

When Commodore planned a follow-up to the VIC-20, there was new uncertainty about the company's direction: Should it aim for the business or the home market? In April 2022, I interviewed Commodore chip designer Al Charpentier about the process of developing the C64, and he told me how this uncertainty played out in the company and led to Peddle leaving:

> **Al Charpentier:** Chuck Peddle believed that the only market for computers was business. When the VIC-20 proved that there was a home market, the whole focus shifted. I had started out designing the VIC-II as a better VIC-I for games. Halfway through that design it became clear that it was meant to be a computer.[57]

In early 1981, Charpentier was tasked with developing from scratch these new "state-of-the-art video and sound chips for the world's next great video

game," initially meant to be sold to other manufacturers.[58] Charpentier and his team examined and borrowed ideas from the current game machines.[59]

This was still early in the history of integrated circuits—chips—and Charpentier has described how the limited tools of the time meant that the design process had very physical stages, with a "pen plot that was literally 10 feet by 10 feet. And you would crawl around on the floor, making sure all the connections were right. It's amazing that it worked."[60] The physical aspect of the work also applied to the chips on a smaller scale. The sound chip was originally planned to have four voices, but it became slightly too big, and one voice was removed because a smaller chip would be cheaper to produce.[61]

In the top middle of the circuit board is the 6510 CPU—the central processing unit that runs the programs of the machine. Bill Mensch from the 6502 development team told me emphatically that the 6502 and other chips they designed were not meant for computers but for controlling smaller devices.[62] For Commodore, using an in-house CPU chip was an obvious choice, bringing the economic benefits of vertical integration, but for others the 6502 had the advantage of being cheap, and Wozniak famously lined up to buy it from Chuck Peddle.[63] The 6502 was inspired by the more expensive Motorola 6800 chip but simplified in several ways, limiting its math abilities and the complexity of the programs it could run.[64]

Top left on the circuit board are the two 6526 CIA chips handling most input and output through connectors for tapes, floppy drives, and printers. The C64 also has a serious side. For machine language programs, the Kernal (spelled with "a") has well-defined routines and specified entry points, like a modern operating system. The serious C64 is nearly identical to a VIC-20 or PET computer before it. It is just a display of 40-by-25 characters, each taken from the standard character set, with the ability to load and save data to tapes or floppy disks.

A Home Computer for Arcade Games and Beyond

It is now difficult to grasp just how central arcade games were in the popular imagination of the early 1980s. Figure 0.11 shows popular arcade

games of 1979–1981. (► See games on the website.) If you were designing a computer for playing games at this time, you would be looking at these. These games are all two-dimensional, seen from the side or the top (*Pac-Man* combines both perspectives). They consist of moving objects—ghosts, aliens, barrels, cars—most of them on static backgrounds, and the gameplay centers on whether objects collide. *Rally-X* also scrolls: the world is larger than what the screen can show, and the game pans around the world as players move around it. According to chip designer Al Charpentier and Commodore game developer Andy Finkel, such games and their imagined evolution played a key role in the design of the C64:

> **Al Charpentier:** The kind of games that I was envisioning [were] more like Nintendo's, even though the Nintendo Entertainment System didn't exist yet. But like the *Zelda* series. That kind of game. That required a lot more action on the screen than you could do with current technologies. That's why I spent so much time on the sprites, trying to create that environment. It was the first step. At the time, the VIC-II was very advanced, but it opened the door for a lot of other things.
>
> **Jesper Juul:** Right. Arcade games have to be very short. Epic games clearly have to be in the home.
>
> **Al Charpentier:** Exactly. You can't you get a quarter each time. I was more into longer play games, like Zelda, which came out three years later. I wasn't the only guy thinking that way.[65]

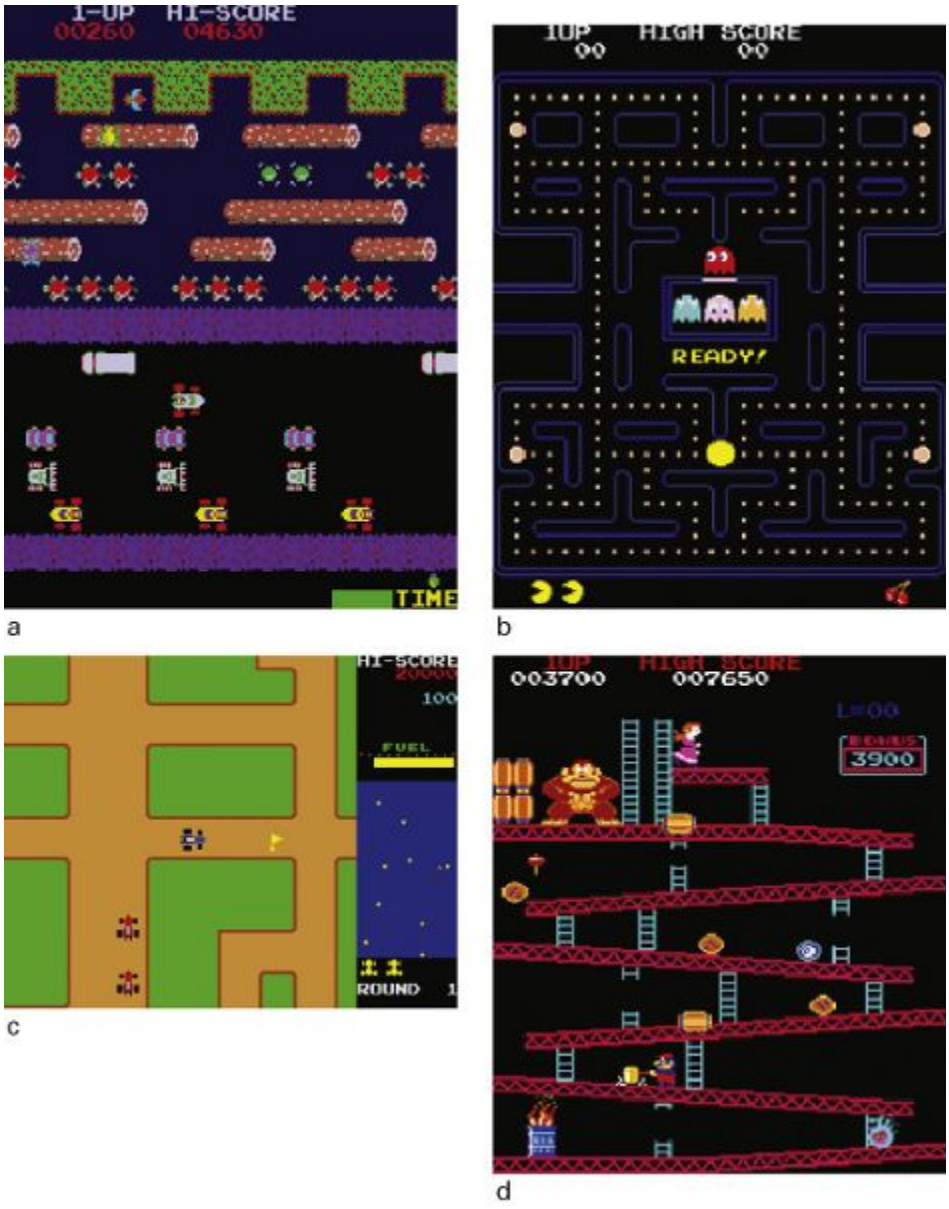


Figure 0.11 Arcade games popular in 1981 when the C64 was being designed. a) *Frogger* (Konami, 1981); b) *Pac-man* (Namco, 1980); c) *Rally-X* (Namco, 1980); d) *Donkey Kong* (Nintendo, 1981). All are two-dimensional with moving objects against a static background. The background scrolls in *Rally-X*.

Finkel explained how the software development team regularly played arcade games and requested that Charpentier put the relevant features in the chip. I asked him if the C64's abilities coincided with the games he wanted to make.

> **Andy Finkel:** Pretty much. The game writers at the VIC group got to talk to Al [Charpentier] and tell him what we liked about the VIC-20, what we didn't like, and what we wanted. We mentioned player-missile graphics [sprites], which we missed because it would have been nice not to have to handle every collision in software, and to move objects more easily. The game people were also oriented towards arcade games. We would go out to lunch and to an arcade fairly frequently and play games.[66]

To understand the C64's capabilities, it is worthwhile to understand not only the popular games at the time it was designed but also the technological pressures facing a video game graphics chip designer in 1981. How do you move the frog, the car, or the spaceship across the game world? In 1981, there were three main problems to deal with: how to keep track of all pixels on the screen within the memory (RAM) available, how to update the screen fast enough, and how to make movement smooth.

Figure 0.12 shows a simple C64 screen with 320 columns and 200 rows of pixels, a balloon sprite, and some text. It figures that the computer must in some way keep track of the pixels on the screen, and we can imagine an internal graph paper that the computer keeps in memory. The problem is that a 320-by-200 pixel image in full color would use far more memory than is available in the C64.[67] We must scale it back. The C64 has a simpler "bitmap" mode where the screen is divided into smaller 8-by-8 areas, where each pixel can be one of two colors, but to make functional games we can use a predefined set of characters for both text and games (which we will get to), and the screen with its 40-by-25 characters now takes only 1 KB of memory, which is manageable on the machine.

Figure 0.12 Simple C64 program demonstrating the platform's core elements: a default light-blue border around a central rectangle of 320-by-200 pixels, text/characters, and a *sprite*—a moveable graphics object—representing a balloon on top.

What about movement? If we think of changing every single pixel by hand on the graph paper when we need to move the balloon sprite, we sense why updating the entire screen would be too slow with 1981 technology. Using this approach, a computer of the time—like a ZX Spectrum or Apple II—could only draw a few small objects fast enough, and in practice often flickering. The VIC-II chip solved these challenges by combining moveable sprites, character graphics, a special scroll feature, and raster synchronization features for smooth movement. Figure 0.13 shows the simple layered construction of the C64 display, with a background, characters on top, sprites on top of that, and the border at the very top. It may sound abstract, but it is in fact very concrete, based on a few core principles:

Sprites: To make action games feasible, the C64 designers added sprites for displaying objects in motion. Sprites are small moveable images, so named because they float above the background. The video chip allows programmers to position eight individual sprites with no need to redraw the underlying pixels, like moving puppets on strings.[68]

Characters: The character layer is divided into 40 columns and 25 rows, and each character element can be one of 256 characters. Figure 0.14 shows the two standard C64 character sets, the uppercase PETSCII character set

with many graphical characters like lines, suits, and circles, and the lowercase-uppercase set, usually used for writing. The PETSCII graphical characters were created both to compensate for the lack of bitmap abilities on the early PET machines, and because Chuck Peddle and Jack Tramiel enjoyed blackjack, and the machine therefore needed to display card suits.[69] On the C64, developers can also redefine the character set themselves to represent game elements such as mazes and landscapes. (On other systems, characters are sometimes called tile graphics.)

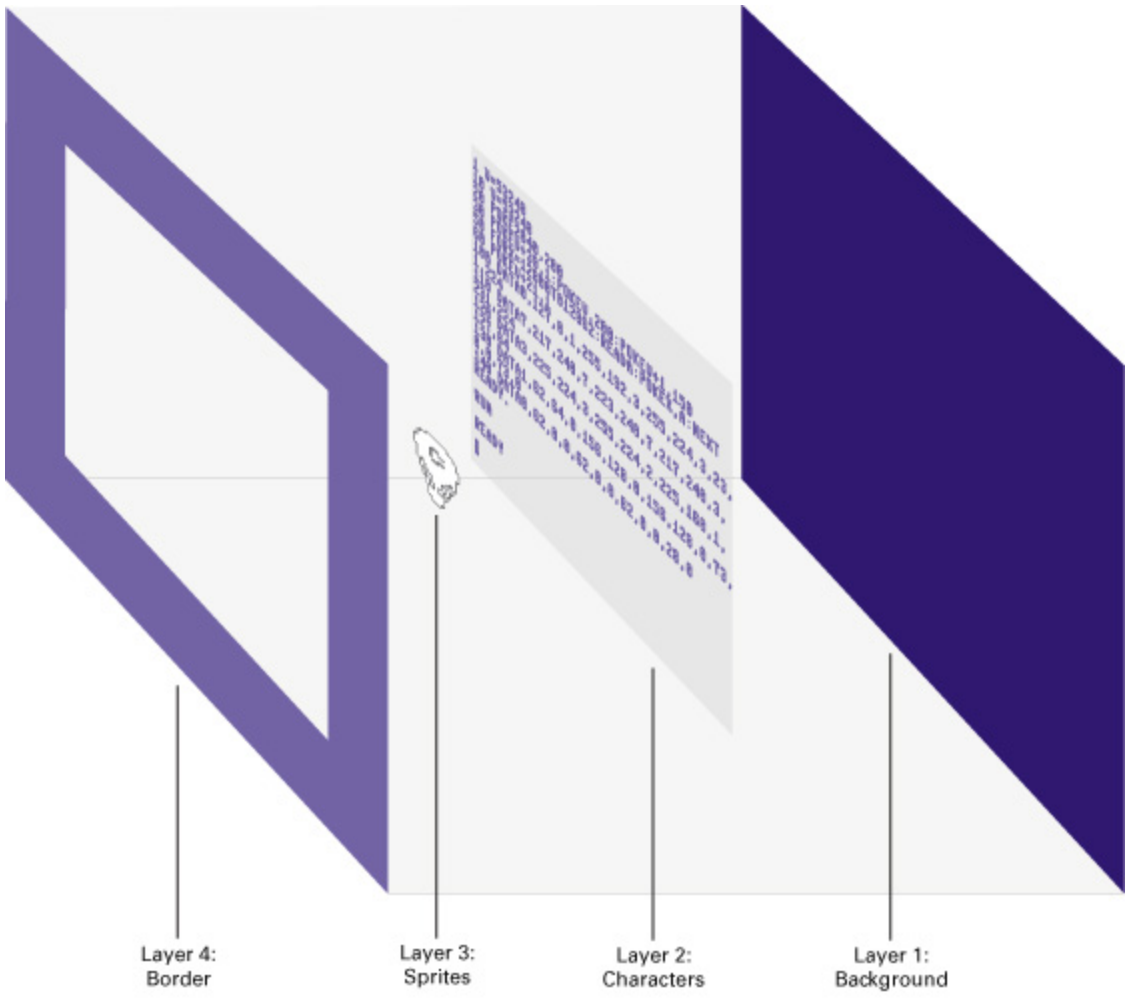


Figure 0.13 Illustration of the four layers of the C64 display with 1) a background (dark blue); 2) text/bitmap (light blue); 3) sprites (white balloon); and 4) a border (light blue). Text and sprites can be changed and moved independently. The border can be expanded to cover ("blank") the entire screen, and through programming tricks it is also possible to hide the border, showing background and sprites instead. Sprites are small images that can be moved on top of (or below) the text or bitmap layer.

Hi-res or multicolor: All these graphics types on the C64—bitmaps, sprites, characters—can be shown in either monochrome hi-res or multicolor. In multicolor, the horizontal resolution is halved to 160 by 200, but in return, we can change the color of the individual pixel to one of four different colors. Figure 0.15 shows a hi-res character (8-by-8 pixels in two colors) and a multicolor character (4-by-8 pixels in four colors).[70] This is the general C64 tradeoff, between high resolution and many colors.

Figure 0.14 The C64 can switch between two built-in character sets. The uppercase-only character set contains more graphical PETSCII characters including card suits, but the upper/lowercase character set is better for writing. The PETSCII characters allow users to create simple graphics directly on the screen. All characters exist in regular and inverted variations. Users can also define their own character sets.

Figure 0.15 Hi-res and multicolor characters. Hi-res gives higher resolution but only two colors, while multicolor halves the horizontal resolution in exchange for four colors.

Sixteen colors: What colors can be used then? The C64 can show exactly sixteen different colors and no more, a fixed palette permanently inscribed in the C64 chips. The colors are mysterious and subtly dependent on both display (figure 0.16) and the specific C64 model.[71] The colors are numbered from 0 to 15, and a C64 user is as likely to know them by number as by name: 0: BLACK, 1: WHITE, 2: RED, 3: CYAN, 4: PURPLE, 5: GREEN, 6: BLUE, 7: YELLOW, 8: ORANGE, 9: BROWN, 10: LIGHT RED, 11: GRAY 1, 12: GRAY 2, 13: LIGHT GREEN, 14: LIGHT BLUE,

15: GRAY 3.[72] *Why these colors?* Attending a 2022 panel of Commodore engineers, I asked Al Charpentier about the reasoning behind the colors:

> **Jesper Juul:** I understand the limits of the four bits leading to the sixteen colors, but how were the sixteen colors decided upon? [Audience laughter]
>
> **Al Charpentier:** It was somewhat serendipity. There were certain ratios that would work. Honestly, this was sort of driven by the hardware. I had to choose resistors with enough separation from one to the next. And from a manufacturing perspective, they would always be about the same. I knew I needed the primaries, so I had to fix those. And then after that, it came down to what was left.[73]

Figure 0.16 The sixteen colors that the C64 can show—on a modern emulator (left) and on an old screen (right). On modern monitors and emulators, colors are clearer than they were on old TVs.

Though C64 colors were limited, they did have the advantage of *actually working* in most of the world, whereas competing Apple II and Atari 8-bit computers were designed for the NTSC standard, and color therefore initially did not work outside North America (and select other countries).

Scrolling: But what about *Rally-X* (figure 0.11)? For such a game, we need to scroll the world, like panning a camera around a larger universe. Again, if we imagine redrawing each pixel of the background by hand, we can understand why this would be too slow. The Commodore 64 deals with this using a special scroll feature, where we can instruct the VIC-II chip to

move the character layer (figure 0.13) horizontally or vertically one pixel at a time within an 8-pixel range, allowing for smooth scrolling games.[74]

Raster: Even if we can move all these graphics quickly, they will flicker ("tear") if we change them while they are being drawn by the TV's raster beam. To enable smooth movement and prevent flickering, the VIC-II chip lets programmers synchronize their programs with the position of the raster beam that draws the TV image.

Sound: The C64 was known for its music, and some magazines even featured C64 music charts, but how did a game computer become such a platform? Bob Yannes, designer of the C64 SID sound chip, believed that existing sound chips "were primitive and obviously had been designed by people who knew nothing about music," and he wanted to make chips for a state-of-the-art music synthesizer.[75] The SID has only three voices, which posed the challenge that most Western music requires at least four voices—three to set the harmony and one for the melody. Games also often need to use one voice for sound effects. This decision to cut the number of voices from four to three, driven by a tight schedule and monetary considerations, inadvertently led to a distinctive hyperactive C64 musical style with common use of fast notes—arpeggios—to establish a song's harmony. I examine sound and music in further detail in chapter 3.

Launching the Commodore 64

The C64 was announced at the Consumer Electronics Show in Las Vegas in January 1982 for a price of $595 and launched in August the same year. Competitors were surprised at the machine's low price point, made possible by Commodore's chip manufacturing capabilities.[76]

Reflecting Commodore's history as a manufacturer of office equipment like typewriters and calculators, and unlike the competing TRS-80 and Apple II machines, consumers were not meant to open Commodore's PET, VIC-20, and C64 computers, as they were designed with support chips for communication, as well as ample connectors, including the special bus connector that—much like USB connectors today—allowed for connecting multiple disk drives and printers to the same computer port.

Unfortunately, the C64 disk drive was excruciatingly slow. Engineer Bob Russell had designed the C64 circuit board with extra connections, or

traces, from the disk drive port to the 6526 chips to support a fast disk drive, but when the circuit board was put into production, production engineers had removed the connections. Hundreds of thousands of boards had already been made, so modifying them was out of the question, and the 1541 disk drive ended up much slower than it needed to be.[77] Obviously, Commodore rectified this in the next update, right? No, Commodore did not, in part because the company quickly set its sights on newer computers. Still, the flaw of the slow disk drive had strangely positive downstream consequences for the C64, creating a cottage industry of aftermarket upgrades.

Figure 0.17 Actor Danny Kaye playing *Tic-Tac-Toe* against the "Bertie the Brain" computer in 1950. The history of computers is full of games, useful for demonstrating to the public that computers could solve problems traditionally solved by humans. (Photo by Bernard Hoffman/The LIFE Picture Collection/Shutterstock.)

Fear of Games

With a computer designed in part for video games, a question arose: *What if the Commodore 64 is too much fun*? The question of games ran through the history of early computers, and an iconic 1950 *Life Magazine* photo (figure 0.17) shows actor Danny Kaye defeating the "Bertie the Brain" computer at *Tic-Tac-Toe*. In the 1950s and '60s, public demonstrations of computers often featured *Tic-Tac-Toe* or *Nim*, the game of taking turns removing matches.[78] Games and computers have always displayed an affinity, because the rules of games are exactly the kind of problem that computers have historically been good at dealing with.[79] Games also made early computers comprehensible to the public because we understand the problem being solved, and because the computer takes a human role, hence the common description of early computers as "electronic brains."

When computers were first marketed for home use in the late 1970s, this coincided with the initial popularity of arcade games and the first popular home video games. Video games helped sell home computers, but many felt that they made computers seem less useful.[80] At launch, the C64 was the computer most deliberately designed with video games in mind, alongside Atari's 8-bit line. As we will see, neither Sinclair (with the ZX Spectrum), Apple, nor IBM designed their computers with the features for fast-moving action games that the C64 had.[81] Why not? *BYTE* columnist Bruce Webster remarked that IBM "was somehow afraid that making graphics (especially color graphics) a standard feature would 'lower' its system to the level of the Apple/Atari/Commodore computers which most people viewed as game machines."[82] Apple shared this worry, and in 1986 Apple chairman John Sculley III puzzlingly assured the public that Apple devices were not *home computers* but computers for the home:

> Apple Chairman John Sculley emphatically denied that his company is selling home computers. Apple, he maintained, sells "computers for use in the home." The distinction, supposedly, is that a home computer is a low-powered, low-end machine primarily suited for playing games, and that a personal computer is more practical and pricey.[83]

Steve Jobs later criticized the early Commodore Amiga computer for having "too much hardware," implying that computers should not have dedicated hardware for updating graphics, much less for video games.[84]

Why couldn't games, word processing, and other serious software coexist? Technology historians Natale and Balbi explain how, when a medium or technology is new, it is subject to competing imaginaries (Is the computer a typewriter? A calculator? A terminal? A circuit board? A game device?) and can be understood in many ways. This has a name: *interpretative flexibility*. As a medium or technology later becomes entrenched, a consensus grows about what the technology is for, and many early ideas and possibilities become sidelined.[85] Commodore and other manufacturers were worried that a computer seen as a game machine would lose this interpretative flexibility and be judged as only useful for games, making it difficult for consumers to justify the purchase.[86]

Commodore therefore had to navigate the game question. In a 1981 TV commercial for the VIC-20 computer, a young man applies for a job based on his video game prowess but is rejected with the advice: "If you're going to spend your time playing video games, why not play them on something that can also teach you about computing? Get a Commodore VIC-20. It's tough to grow up in a computer age without learning about computers."[87] While the VIC-20 was advertised as a video game device that could teach consumers about computing, the Commodore 64 advertising shifted its emphasis and mostly promoted the C64 as a serious computer that could *also* be used for playing video games. Early print ads for the Commodore 64 promoted the genuinely low price of the machine compared to competitors and emphasized literacy and productivity but mentioned games only as an aside.[88]

In some ways, modern computers and phones are all Commodore 64s and Amigas. Though Commodore filed for bankruptcy in 1994, the company won the argument about graphics, sound, and games resoundingly, and the matter is settled today: Since the 2000s, operating systems like Windows, later versions of macOS, as well as modern iOS and Android phones all use dedicated hardware for quickly and smoothly updating the interface and playing sound, and the same hardware is also used for playing games. Despite commentators of the 1980s and 1990s forcefully arguing against "unserious" abilities like color and game support, there is now wide agreement that such features are crucial for both games and modern graphical user interfaces.

Five Imagined Commodore 64s

This is not the first book about a computer, and I should point to the writers who have inspired me. I want to avoid the heroic story of lone hackers, and I am inspired by Tracy Kidder's *The Soul of a New Machine*, which shows how technological development is a meeting of many people and ideas.[89] *Who Are You*, Alex Custodio's book about Nintendo's Game Boy Advance, shows that we can look beyond the officially released hardware, and examine how users can repurpose a device.[90] Laine Nooney's *The Apple II Age* argues that not just hardware but software was central to defining the personal computer.[91] Alison Gazzard's *Now the Chips Are Down* about the BBC Micro shows how cultural forces, such as government policy, can shape a computer.[92] Jimmy Maher's *The Future Was Here* about the Commodore Amiga, and Ian Bogost and Nick Montfort's *Racing the Beam* about the Atari 2600 each analyze the history of a platform, the impetus behind it, and how its technical features are used by developers.[93] Julien Mailland and Kevin Driscoll's *Minitel: Welcome to the Internet* shows that studying a platform can reveal larger insights, such as into the relation between the state and the market in the US and France.[94] The history of the C64 can similarly teach us more about the history of computing and video games. Finally, Morgan Ames's *The Charisma Machine* about the One Laptop per Child initiative shares my interest in the interaction between the ideas and imaginaries about a machine and its concrete hardware.[95]

To write this book, I have tried hundreds of pieces of software, many of them games, I have developed BASIC programs, rewritten machine language demos, and studied computer and game magazines. One goal has been to write a C64 history that includes many different relationships to the C64, from users to hardware designers to developers and product managers. I have therefore interviewed a range of people about the C64, and their voices appear throughout:

- Al Charpentier, Commodore engineer and designer of the C64's VIC-II graphics chip
- Andy Finkel, Commodore 64 software product manager, developer, and coauthor of the *Commodore 64 Programmer's Reference Guide*
- Jesper Christensen, sales director, Commodore Scandinavia

- *Swyx* and *Chix*, members of the Triangle 3532 demo group
- *Trap*, member of the Bonzai demo group, active on the demoscene in 1989 and now
- *Einstein*, member of the Upfront demo group and organizer of a 1989 demoparty
- Raquel Meyers, visual artist and demoscene contributor
- Johan Grip, creator of the C128Neo circuit board

I have also corresponded with VIC-20 product manager Michael Tomczyk, researcher and writer for *Zzap! Italia* Matteo Bittanti, 6502 chip designer Bill Mensch, game developer Noah Falstein, and Professor Stephen Edwards.

The Commodore 64 was never upgraded to have more capabilities, but at different points in time, there were radically different ideas—imaginaries—about what the Commodore 64 was. How many? Aforementioned researchers Natale and Balbi argue that media or technologies go through three different stages, from before the medium exists, to when it is new, to when it is obsolete.[96] Examining the C64, it became clear that there was more to it. Not only were there many different ideas about the C64 when it was new, it has also taken on an entirely new meaning *after* becoming obsolete. I have therefore structured this book around five lives, five imaginaries of the Commodore 64, roughly in their historical appearance. The imaginaries are distinguished by what they are oriented towards: 1) continuing the computer practices of the past, 2) using the computer for its advertised and intended uses, 3) using the computer for its own sake, 4) applying new ideas to the computer as it becomes old, and 5) appreciating the computer as a historical artifact that takes on a new meaning.

For now, I leave it to the reader to consider how this five-step structure maps to other computers or situations, but I will return to it at the end of the book. The imaginaries each appeared and became dominant at specific points in C64 history, but they also overlapped, and most C64 users likely had experience with multiple imaginaries. Each imaginary can also subtly change, as when new game types changed the conception of the kind of games the C64 could play. Each imaginary provides its own answer to a user's question, "What is this computer for?"

Chapter 1: The BASIC Computer for Family, Business, and Education. Turning on the C64 takes you directly into its READY prompt and blinking cursor, and the commands you can use to perform simple actions are identical to the commands you can use for programming.

What is this computer for? Users had been prepared for computers by the media, word of mouth, and advertising. I show how Commodore's advertising promoted the C64 as a general-purpose computer for productivity and literacy, mentioning games only as an aside. The ads divided C64 computer use along strict gender lines, showing women using the C64 only for utility, while presenting the additional uses of programming and game-playing as appealing to men. Hence, the chapter also follows how games and programming came to be construed as male activities.

With an only recently developed software industry, typing in BASIC programs was common, and the built-in BASIC language of the C64 connected the machine to a long tradition of BASIC—and BASIC game—programming. I examine the *Commodore 64 User's Guide* to show its assumptions about what users would do with the C64 and how the limitations of C64 BASIC finally pushed users to write in the faster but more technical machine language.

Chapter 2: The Arcade Game Machine (and Beyond). Arcade games were a target for the design of the C64, and the promise of early home computers was often to "bring the arcade home." The history of C64 games is the history of first emulating the physical arcade games, then replacing arcade games as the dominant type of video games, and then figuring out what video games should be once in the home. I examine how the C64's design enabled developers to move away from arcade game conventions, focusing instead on high-quality graphics, surreal settings, extended playtimes, and making games that were entirely outside conventional genres, such as the original *SimCity*.

While the C64 was the only mid-1980s computer popular on both sides of the Atlantic, I show how US developers gradually lost influence in Europe from 1982 to 1990, and I examine the European tradition of platform and action-adventure games, predating both *Super Mario Bros.* and *Castlevania*. I also discuss how the scrolling facilities of the C64 led to

a specific C64 game tradition that was central to the C64's significant—and often forgotten—influence on video game history.

Through cheats, users were faster than developers to realize that GAME OVER was unnecessary in home games. The culture of modifying programs and the role of piracy on the platform—including software piracy from Commodore itself—had the side effect of making programming attractive to users and made even game-oriented magazines publish technical articles.

Chapter 3: Against Intentions. The scrolling facilities central to many games on the C64 had two exploitable glitches with significant cultural ramifications: the border glitches. In 1985 and 1986 it was discovered that it was possible to show graphics in the normally light blue border of the machine. These and other glitches were central to the C64 demoscene, a loosely connected culture, mostly in Germany and northern Europe, where users would meet and show off their graphical demonstration programs. I examine the demoscene by looking at a specific "demoparty" in December 1989. I show that demoscene materials are dominated not by technical discussions but by gossip about "demogroups" and members and by jockeying for social status in charts. I also show that the demoscene often used public support such as spaces provided by municipalities for nonprofit organizations.

Common demoscene tricks are unintended consequences of the C64 design goals, and these tricks supported a culture around the C64 and helped create a new imaginary as a device for subversive computing. Contrary to my initial expectation when writing this book, computer magazines retained a focus on programming until the very end, with game magazines reporting breathlessly on the demoscene while commercial software development died out.

Chapter 4: Keeping Up with the Future. The C64 was launched in 1982, but newer machines quickly arrived to challenge it. There was a worry that the C64 was falling behind, but a new graphical interface, GEOS, was made. Developers worked valiantly to keep the C64 a contender against newer computers, in its interface, in productivity software, and in games. Commodore Amiga games like *Defender of the Crown* and *Lemmings* were ported to the C64, with reviewers anxiously reporting whether the C64 version could live up to the original.

Chapter 5: The Eternal Commodore 64 Style. Decades later, the C64 is a historical machine, whose quirks and limitations no longer represent cutting-edge technology, as the urgency of the "keeping up" stage is gone. There are C64 software emulators, recreations, and projects to reverse engineer the machine's hardware. The limitations of the C64 are now charming rather than stifling, a historical style we can use or emulate at will. The question in this final phase is the role of the physical machine. Do you need a C64 to participate in C64 culture? Can you build your own board? Do you need chips from Commodore to make a C64?

I see these efforts not just as nostalgia but rather as bringing the C64 in contact with contemporary culture and computing, using newer tools to develop software on the C64, making the C64 more convenient than it was originally, questioning: Do we always need newer computers?

Finally, I sum up the two main threads of the book: how the unchanging C64 could change through an interaction between imaginaries and the concrete machine, and how the C64 has been ignored in history because it was *too much fun*.

How to read the book: The story of the C64 combines advertising, chip designers, computer culture, users, magazines, games, and programming. I focus on the main narrative in the text and elaborate technical details in notes and text boxes. All these perspectives matter equally, but depending on your interests and personal experience with the C64, you can read the book in three ways:

1. If your main interest is in the culture and history of the C64, you can focus on the main narrative of each chapter and skim some of the technical discussions.
2. If your interest is in C64 games, focus on the introduction, chapters 1, 2, and 4, and watch the videos on the website.
3. If you are interested in the technical tricks of games and demos, focus on the introduction and chapters 1, 2, 3, and 4, and try the sample programs and source code on the website.

1

The BASIC Computer for Family, Business, and Education

Imaginary #1:	The Commodore 64 is a universal computer for the whole family, for business, education, and for learning about computers.
	The Commodore 64 is part of the BASIC programming tradition.
Programming Language:	BASIC
Graphics:	Standard character set
Colors:	Dark blue, light blue
Quotes:	Commodore 64: Multi-purpose personal computer for educational and business use! *—C64 European box, 1982*
	Commodore 64: The Complete Computer System. Personal Business Aids. Communication. Education. Plus Entertainment.

	—C64 US box, 1982
	The COMMODORE 64 USER'S GUIDE is designed to give you all the information you need to properly set up your equipment, get acquainted with operating the COMMODORE 64, and give you a simple, fun start at learning to make your own programs.
	For those of you who don't want to bother learning how to program, we've put all the information you need to use Commodore programs or other prepackaged programs and/or game cartridges (third party software) right up front. —Commodore 64 User's Guide, *1982*
	But even today some amazing games are possible with the current breed of home computers. That's what this book is all about: games for home microcomputers, minis, timesharing systems and even large mainframes—in short, any computer that speaks Basic. *—David Ahl's* BASIC Computer Games: Microcomputer Edition, *1978*
Examples:	jesperjuul.net/c64/1/

It begins with a box and inside it, the C64. The US box (figure 1.1) announces a “personal computer with professional power,” while the European box promises a “multi-purpose personal computer for educational and business use.” In this chapter, I will look at the early days of the C64 including how it was advertised, examine how its BASIC programming language connected to a longer BASIC tradition, and analyze the experience of opening the *User’s Guide* and beginning to program.

The early business of computer selling came with a giant advantage: *everybody knew that computers were the future*, but the boxes nevertheless needed to be elaborate, because users could not be counted on to know what a computer was for.[1] A 1982 *New York Times* article titled “The Home Computer Arrives” notes that, “while many people are buying computers to learn about them, analysts say the market’s continued growth will depend on finding practical uses.”[2] Box design and advertising are deliberate attempts at creating imaginaries—at giving potential buyers positive ideas about what a product is, how it fits in the world, who it is for, and what it will do for them.[3]

Figure 1.1 Top: 1982 US C64 box at launch, emphasizing the use for business, communication, and education “plus” entertainment. Bottom: European C64 box at launch, emphasizing education, home organization, science, and business.

Even though computers in the home were new, the boxes tried to bill them as efficient ways to perform well-known activities (education, budgeting, communication, office work)—activities with predetermined target audiences that would dictate where a computer should be sold and to whom.[4] We can think of several traditions (see figure 0.7) leading to home computers, each carrying their own set of assumptions:

Mainframe computer terminals: Many university graduates of the 1970s had worked on timesharing terminals during their studies and were

looking for a similar experience at home.[5] If home computers were part of the computer terminal tradition, the home computer should be sold to recently graduated consumers, rather than to the general public. Supporting this, the cover of the July 1976 issue of *Popular Electronics* promoted the SOL-20 computer as an "intelligent terminal." Given that many early computer programmers were women, from the ENIAC programmers to Grace Hopper and Margaret Hamilton, we would expect that home computers—and programming—would also have been marketed to women.[6]

Electrical engineering: The first microprocessor-based computers for purchase—like the Micral, JOLT, KIM-1, and Apple I—were marketed to engineers. *Radio-Electronics* magazine published Don Lancaster's TV Typewriter hobbyist project in 1973, a mostly impractical device which could only put typed characters on a TV screen but which nevertheless influenced the development of home computers.[7] In the electrical engineering tradition, a computer was just the end result of an electronics project and should be sold alongside other electronics and electrical engineering products.

Calculators: Intel's first microprocessor, the 4004, was developed for a calculator company and only later marketed as a more general unit. Similarly, Commodore sold calculators and bought MOS Technology intending to build calculators and only later expanding to computers.[8] Olivetti's 1965 Programma 101 (figure 0.7), sometimes called "the world's first personal computer," is an early example of a calculator that can be programmed.[9] If home computers were part of this tradition, they were calculation devices and should be sold where office supplies and machinery were sold.

Typewriters: Commodore had started as a typewriter company. Unlike electrical engineering, typing was seen as a feminine activity handled by secretaries.[10] The 1981 Xerox Star, an early computer with a graphical interface, was promoted in a brochure as "designed specifically for professional businesspeople with little or no typing skills."[11] Adele Goldberg of Xerox recalls, "We had it all structured so they would never touch the keyboard. It was all mouse-pointing and mouse-clicking, because we knew these guys wouldn't type. In those days, that wasn't macho."[12] If

home computers were a new kind of typewriter, they should be sold as office equipment, and in part targeted at women.

Video games: Finally, a recurring idea was that computers were video game–playing devices. Computers were always used for playing games, but many home computers were unique in having been explicitly designed for the purpose. If home computers were part of this tradition, they should be sold in toy and TV stores, alongside other kinds of entertainment.

Learning from Advertisements

How did Commodore advertise and sell the C64 then? Interestingly, there was no official global strategy, as each subsidiary or country distributor would procure its own advertising, but there were distinct patterns to the marketing of the C64.[13] To understand the initial promotion of the C64, we examined fifty-six extant television ads for Commodore computers. (► See ads on the website.) These ads were aired across multiple countries (primarily the US, UK, and Australia). Our sample is likely different from the commercials that have been lost to time.[14] We wanted to understand the role of games in advertising and how advertising described the computer's possible uses and users. We also examined *Compute!'s Gazette*, *BYTE*, and German *64'er Magazin* for advertisements for Commodore computers.

The overall picture is that most early C64 advertising was broad-minded in that it showed many different uses for the computer and a wide variety of users. Yet the advertising divided computer use into separate categories for men and women. Where women would be assigned brief interactions with the computer as part of other roles (teaching, retail, or housekeeping), only young men were shown as having sustained computer engagement with activities such as programming or playing noneducational video games.

Let's examine one example: the Australian TV ads for the Commodore 64 known as the "Keeping Up with the Commodore" commercials. These advertisements feature an upbeat and catchy jingle with the refrain, "Are you keeping up with the Commodore? Because the Commodore is keeping up with you," a slogan that positions the Commodore 64 and the prospective computer user as collaborators in a cycle of mutual inspiration and innovation. It is a cheerful collaboration with an unmistakable undercurrent of anxiety, as failing to "keep up" with Commodore's products

leaves you at risk of falling behind technologically and, several ads imply, academically and socioeconomically. The Keeping Up with the Commodore commercials are structured variations around the following: a person interacts with a Commodore 64, then turns to the camera, smiles, and salutes. These clips of Commodore users are intercut with footage of people (typically young women) enjoying themselves outdoors, often without the presence of a computer of any kind.

Figure 1.2 1983 Australian Keeping Up with the Commodore commercial: On the left, a woman with a pink spreadsheet uses the computer only fleetingly; on the right, a teenage boy is deeply engaged with the computer, programming in BASIC, and preparing for his future career. The stacks of paper signal that he is a programmer.

Nearly all Commodore 64 users in these commercials are white businessmen, and the two women who are shown using a computer do not appear to be typing. The first, a young woman in a clothing store, presses a single button to populate a cell in her (hot pink) spreadsheet before turning away from the monitor (figure 1.2). The second, an elementary school teacher, presses a key twice to bring up an image of Saturn for her class. Although both women are portrayed as confident computer users, they do not interface with the machine itself for more than an instant before turning their backs to the computer and smiling. By contrast, several of the male computer users spend many seconds typing, focusing intently on the monitor, and only turn to face the camera after they have completed their more complicated work.

The most glaring example of this discrepancy can be found in the Keeping Up with the Commodore commercials that feature a prolonged closeup of BASIC commands rapidly climbing up a monitor.[15] The camera zooms out to reveal a teenage boy typing away at his Commodore 64, pen and paper at his side (figure 1.2). A book titled *Background to Careers* rests on top of his monitor, implying that he is professionally minded and that his computer use is a means of preparing for his financial and vocational future. Every element of his environment is conspicuously serious: he is surrounded by monotone colors, his desk covered in white books and papers. The room is dark, suggesting he is a hard worker, burning the midnight oil to improve his programming skills. He has stacks of paper, which at the time denoted programming. Depending on the commercial, this sequence immediately follows either the clip of the young woman with a pink spreadsheet, her brief tap on the keyboard and colorful surroundings appearing the more frivolous by comparison, or footage of two teenage girls rocketing down a water slide in pink bikinis.[16]

This pattern is clear, but given the five traditions I discussed previously, it is not obvious why it would be so. With the association between typing and women, wouldn't women be the primary users of computers? Given the number of women programmers in early computer history, why wouldn't Commodore market to women when advertising the programming features of its computers? In the ads for the VIC-20, the C64's 1980 predecessor, the intended user was usually a boy, who viewers are told will benefit professionally from having a VIC-20. Yet early internal Commodore documentation of the time made no references to the gender of users, only identifying the initial market as hobbyists.[17] VIC-20 product manager Michael Tomczyk described the advertising focus on male users as having appeared by circumstance during the creation of the ads:

> **Jesper Juul:** Were there reasons to believe that the primary market was men, or was this something the ad agencies decided?
>
> **Michael Tomczyk:** Several key members of our software and engineering teams were women. Our ad agency selected the on-air talent. Our PR and ad directors were women. Featuring males in ads was not intentional.[18]

Color and Games, the Anti-Features

The Australian Keeping Up with the Commodore commercials conspicuously do not mention video games. Of the other ads we examined, the majority reference the C64's ability to play games only as an aside. Studying the 1970s and 1980s home computer market, UK researcher Leslie Haddon notes that games were imagined as having the power to alter children's development, turning them into isolated, antisocial malcontents, desensitizing them to violence, and rendering them helplessly addicted to machines.[19]

If Commodore wanted to sell a C64 family computer, it would ideally position its product as unique in its ability to improve and protect, rather than harm, a child's mind. One tactic for this was to differentiate Commodore computers from video game machines. As Haddon argues, many parents bought microcomputers "at the request of children," and Commodore computers thus had to appeal to the child wanting to play video games, while simultaneously reassuring parents that computers were essential for children's success in a rapidly changing world.[20]

A 1984 Italian TV ad follows the template, explaining that the new Commodore 64 can be used at work (adult man), for studying (adolescent male), and at home (adult woman), as well as for "*giochi, giochi*!" ("games, games!").[21] It also asks viewers how they can manage without a Commodore 64, implying both the practicality of the machine and the importance of not falling behind technologically. Counter to this, the "*giochi, giochi*!" exclamation hints at the pure joy of a new kind of entertainment.

This belief in the double audience for the C64—a young primary user and a parent needing justification for the purchase—carried on to Commodore's later products. UK managing director David Pleasance explained that the logic behind a 1989 Commodore Amiga package combining "trendy games" with productivity software was "to help justify to the parent why they should spend £399 in the belief that their children will get long-term educational benefit from this."[22]

The Commodore 64 in Print: "Do You Know Where You Are?"

The tactic of inducing, and remedying, anxiety about technological change was executed more subtly in print ads. A 1985 *Commodore Microcomputers* magazine ad for the C64 attracts the attention of anxious parents by breathlessly mimicking the infamous "Do you know where your children are?" public service announcements employed on American television in the 1960s through 1980s. Underneath a series of eight photos depicting members of a family using the computer throughout the day (figure 1.3), the ad copy speaks to a wife and mother: "It's 6 a.m. Do you know where your husband is? It's 8 a.m. Do you know where your daughter is? It's 11 a.m. Do you know where you are?" accompanied by the headline, "We promise you won't use the Commodore 64 more than 24 hours a day."[23] The ad does not deny that computers are addictive but capitalizes upon a mother's assumed concern for her family's well-being. Because the Commodore 64 is endlessly useful and appealing, the entire family will remain safely and happily within the home, day and night.

The repetitive nature of the images further reinforces the computer as a comfortable and predictable part of a family's life. The C64 can disrupt and transform lives, but it will meet the unmet needs of each family member without damaging sacrosanct familial structures. This strategy could also be found with other contemporary computer manufacturers, showing an industry desire to integrate home computers into every facet of family life.[24]

Similarly, a 1984 German print ad (figure 1.4) in *64'er Magazin* asks, "Who introduces you to the wonderful world of science?—Commodore Computer."[25] It extols how you can learn astrophysics, biology, chemistry, and languages. You can also organize your data, addresses, or record collection. You can even use it for playing video games, it says at the end.

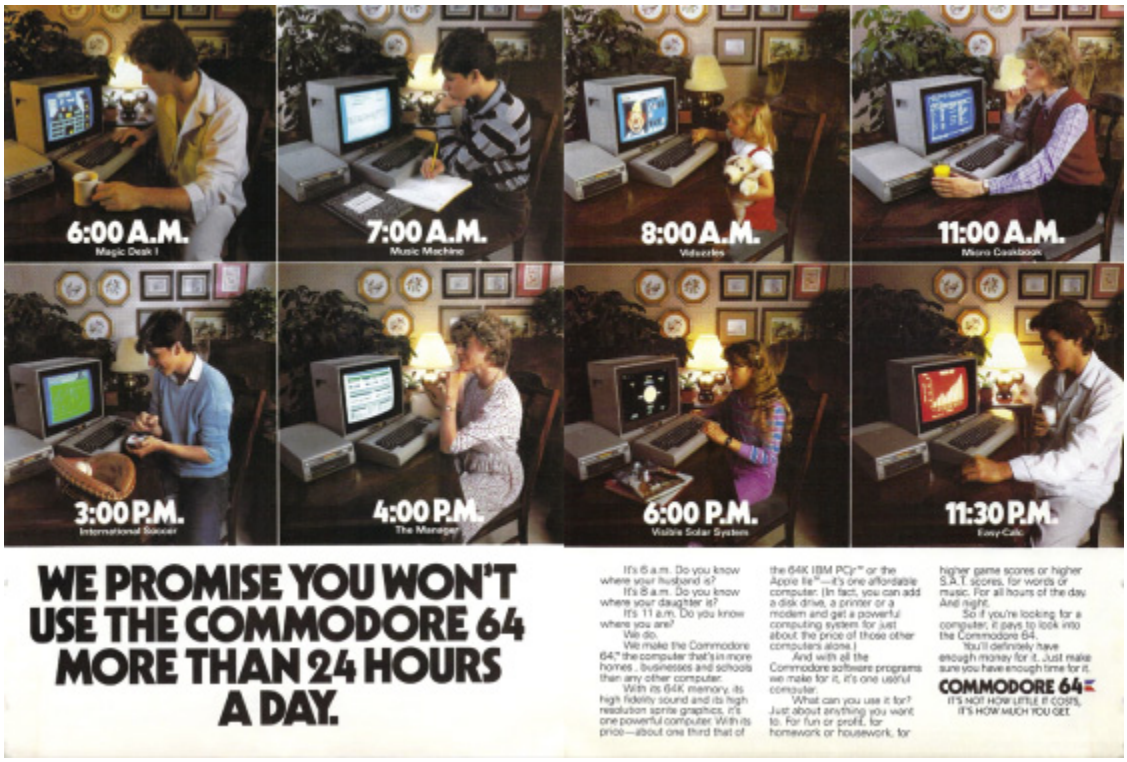


Figure 1.3 Mimicking the US "Do you know where your children are?" public service announcements, this 1985 C64 ad addresses the mother of a household, assuring her that the addictive nature of the C64 will keep family members home and safe.

Distribution Channels

Each of the five traditions I pointed to as preceding the C64 had its own assumed audience, use, and distribution channel. However, the ads were keen to point to many different uses rather than settle on one. Interviewing Jesper Christensen, sales director of Commodore Scandinavia, I expected to hear specifics about their detailed audience segmentation strategies. Instead, he self-effacingly described the company's strategy as "old-fashioned Jutlandic business acumen," where the focus was on getting into as many distribution channels as possible, letting the channels define target audience and uses, and occasionally putting pressure on stores that were promoting a competing product. Bookstores sold C64s, as did supermarkets.[26] When a supermarket chain advertised C64s on the same page as whole chicken, Christensen felt that the C64 had reached mass market penetration. Christensen also explained that photography stores were a central distribution channel, as they perceived the visual abilities of the computer

as tied to photography and video. Italian C64s were similarly sold in many channels, and David Pleasance from Commodore UK reports convincing supermarket chains like Tesco to sell C64s, while smaller independent retailers would provide more specific customer support and guidance.[27] Commodore Germany also sold computers in supermarkets, such as the special Aldi C64 sold at the discount supermarket chain. For the US, the consensus is that Commodore had squeezed and alienated smaller retailers by allowing department store Kmart to sell the C64 at a steep discount.[28] By 1986, though, Commodore lost distribution in Kmart in favor of Nintendo, and US C64 distribution was thus much narrower than European distribution.[29]

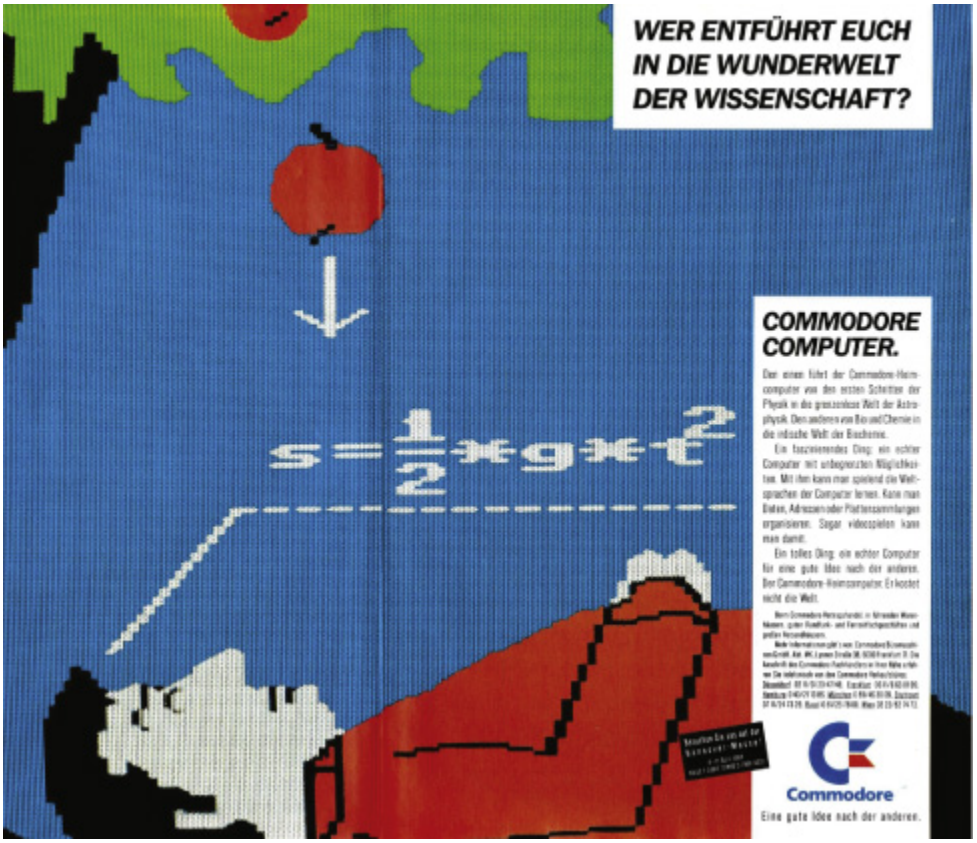


Figure 1.4 Alluding to Isaac Newton's discovery of gravity to emphasize the C64 as a machine for science and learning, this 1984 German print ad asks, "Who introduces you to the wonderful world of science?—Commodore Computer."

The BASIC Tradition

When the C64 was launched in 1982, it was part of a tradition where computers were meant to introduce what Joy Lisi Rankin calls *computing citizenship*.[30] Originally developed at Dartmouth College, BASIC (Beginners' All-Purpose Symbolic Instruction Code) is a programming language meant to make computing accessible. The language was designed from 1962 to 1963 by Thomas Kurtz and John Kemeny as part of a larger vision wherein "computing would be free to students, a privilege comparable to their open stack library access."[31] This also entailed the creation of the Dartmouth Time-Sharing System, which followed the then-new idea of central computers accessible through terminals by multiple users at the same time (figure 1.5). This contrasted with the earlier "batch" model, where a user would submit a program (often on punch cards) and later pick up the results.[32]

Figure 1.5 The BASIC programming language of the C64 was originally developed at Dartmouth College to make computing accessible to all students. “Math professor and future Dartmouth president John Kemeny looks over a program written by his daughter Jennifer Kemeny ’76 using the Teletype computer terminal at their home.” (Photo by Adrian N. Bouchard/courtesy of Rauner Special Collections Library.)

I want to highlight four interesting aspects of this setup. First, Rankin demonstrates how, already at this time—in the mid-1960s—users were communicating, sharing ideas, and sharing programs.[33] As Rankin points out, if we only know the common story of the birth of the home/personal computer ten to fifteen years later—the solo garage genius story—the existence of such early creative computer communities is entirely surprising.

Second, the BASIC programming language is fundamentally pragmatic. Kurtz instructed that, “in all cases where there is a choice between simplicity and efficiency, simplicity is chosen. Every effort will be made to design a system convenient for the user.”[34] This was in contrast with many other programming languages, designed for completeness, elegance, or speed. BASIC programs are meant to be easily readable, to be less fussy about syntax than other languages are, and BASIC does little to enforce good program structure, to the frustration of principled computer scientists.[35]

Third, from the beginning, Dartmouth users used BASIC to create games such as football and bridge. John Kemeny felt that these games served the important function of making users comfortable with computers, “for many inexperienced users the opportunity of playing games against a computer is a major factor in removing psychological blocks that frighten the average human being away from free use of machines.”[36] In this way, games were part of BASIC culture from the very beginning.

Fourth, computer culture of the 1970s was marked by disagreements about whether software should be commercial. In 1976, Bill Gates disparaged hobbyists for using unpaid copies of Microsoft’s Altair BASIC.[37] Gates emphasized that “the value of the computer time we have used exceeds $40,000” but neglected to mention that this computer time was on Harvard’s computer system to which he had free access as an enrolled student. Much of the commercial software industry, in both games and programming tools, grew out of university computer labs.

According to the established mythology, the origins of C64 BASIC, and of the Microsoft company, were Bill Gates and Paul Allen seeing the new MITS Altair 8800 computer on the cover of *Popular Electronics* in Harvard Square in 1975 and deciding to develop a version of BASIC for the machine. The two contacted MITS and offered to develop BASIC for the Altair, and later created versions of BASIC for other computers, notably the Commodore PET, VIC-20, C64, and Amiga, as well as the TRS-80, Apple II, and MS-DOS.[38] There were numerous other implementations of BASIC, including Sinclair BASIC and open-source Tiny BASIC. Companies licensing Microsoft BASIC worked with the Microsoft source code to develop their own versions, in this case Commodore BASIC.[39]

By the time the C64 was being designed in 1981, Commodore BASIC was in version 4 with disk operations and other features, but Jack Tramiel decided that, to reduce costs, the C64 should only have 4 KB of the permanent ROM memory for BASIC. Therefore, the C64 came equipped with the older and smaller Commodore BASIC V2, lacking dedicated disk commands, and users therefore had to master arcane knowledge such as the number 8 signifying the (first) disk drive.[40] Worse, to see the contents of a disk in the drive, users had to write the following:

```
LOAD “$”,8
LIST
```

This erased the current program in memory. To save the BASIC program a user was working on, they had to rely on their recollection of which programs were on a given disk. Furthermore, having multiple consequences, the C64’s BASIC lacked any commands for utilizing the highly promoted sound and graphics abilities of the machine. The limited BASIC 2.0 was an obvious shortcoming, and Commodore never bothered to make updated machines remedying it, instead providing the Simons’ BASIC extension cartridge. The flaw led both to an industry providing the missing commands and to machine language becoming enticing for programmers.

The BASIC Game Platform

The old version of BASIC on the C64 was no problem, though if you only wanted to use older BASIC programs. David Ahl’s 1978 *BASIC Computer Games, Microcomputer Edition* was an update to his earlier book focused on BASIC games on mainframe computers.[41] These BASIC games were freely distributed and used your computer like a teletype terminal—as a line-by-line printer, rather than a screen. Because there are no screens in this kind of BASIC, there are no arcade games, no graphics, and no sound. What was the most popular computer game of 1978? In the book, David Ahl makes the case for the “lunar lander” game genre, “This game in its many different versions and names (ROCKET, LUNAR, LEM, and APOLLO) is by far and away the single most popular computer game.”[42]

In the game, your moon-landing spacecraft descends towards the surface, and you must weigh your velocity against your remaining fuel. Run the retro thrusters at too low a rate and crash, use them too much and run out of fuel (and crash). The game has a storied past and is published in no fewer than three versions in Ahl's book.

```
THIS IS A COMPUTER SIMULATION OF AN APOLLO LUNAR
LANDING CAPSULE.
THE ON-BOARD COMPUTER HAS FAILED (IT WAS MADE BY XEROX)
SO YOU HAVE TO LAND THE CAPSULE MANUALLY.
SET BURN RATE OF RETRO ROCKETS TO ANY VALUE BETWEEN 0
(FREE FALL) AND 200 (MAXIMUM BURN) POUNDS PER SECOND.
SET NEW BURN RATE EVERY 10 SECONDS.
CAPSULE WEIGHT 32,500 LBS; FUEL WEIGHT 16,500 LBS.
GOOD LUCK
SEC MI + FT MPH LB FUEL BURN RATE
0 120 0 3600 16500
? 100
10 110 1190 3436 15500
? 100
20 100 4824 3266 14500
? 200
30 92 2019 2870 12500
? 100
40 84 3549 2679 11500
? 200
50 77 4427 2235 9500
? 200
60 72 1534 1752 7500
? 200
70 68 763 1224 5500
? 100
80 65 572 959 4500
?
SORRY THERE WERE NO SURVIVORS. YOU BLEW IT!
```

► Play LUNAR LEM ROCKET on the website.

Ahl's book assumes a level of technical skill for his users, who are meant to be comfortable typing in programs, identifying the inevitable mistakes, and modifying the program to the BASIC dialect of their platform. The BASIC programming language thus served as a loosely defined platform, in and of itself, like a modern PlayStation or Nintendo Switch. A moderately skilled user thus had access to a large library of games via the book. When Ahl's book came out in 1978, books and magazines were the primary software distribution method.

Did users learn to program to play games, or did they use games to learn to program? The practice of type-in games shows that this is not the right question: games and computational literacy were often entirely intertwined —users learned BASIC programming by typing in games they wanted to play. It was one activity.

A Serious Computer

Did companies really believe that users would be programming their computers? Early on, yes. In the late 1970s, Leonard Tramiel, son of Jack Tramiel, was responsible for reviewing software submitted to Commodore for publication. He has recently admitted to being the person at Commodore to reject the first spreadsheet, VisiCalc, on the premise that users could just program its functionality themselves:

> I said that this piece of software wasn't worth selling because there was nothing that it could do that couldn't also be done by a nearly trivial BASIC program. One reason for having this view was that nearly all of the discussions that I had had with the people involved in producing the first personal computers centered around the primary use being to learn programming. So I was biased to think that everyone that used a PET would be able to write simple programs. The program I'm talking about was VisiCalc.[43]

Figure 1.6 Despite passing over the opportunity to publish the first spreadsheet, VisiCalc, Commodore and software developers took the C64 seriously as a computer for productivity software. Left: Microsoft's MultiPlan spreadsheet (1984). Right: *Compute!'s Gazette*'s type-in word processor SpeedScript (1984).

Leonard Tramiel admits this "inexcusable failure" but contends that Apple had made the exact same call, leading the VisiCalc developers to work with a third-party software publisher. Leonard Tramiel had worked from a common expectation—couldn't users just make the programs themselves?

The later success of commercial software companies took many by surprise. Around 1980, an industry for home computer software emerged, especially in the US.[44] Laine Nooney explains how VisiCalc made the home, or personal, computer palatable to financial analysts and others for whom complex calculations were a central activity. But if the relevance of commercial software had been unclear to Leonard Tramiel in the late 1970s, it was entirely clear to Commodore at the 1982 launch of the C64.

We saw the C64 marketed as a computer for the whole family, for mostly serious purposes and for education, but what did the Commodore company really think? Was the productivity software just a cover for a game-oriented machine? Different sources at Commodore explain the situation quite differently. Software product manager and developer Andy Finkel told me that games were the de facto primary type of software for the C64.

> **Jesper Juul:** Did you experience a push or pull towards games? Both in your own development and as a software manager?
>
> **Andy Finkel:** We were pushing games. That was what sold on the VIC-20. And that's what did well on the 64. It only had 40

> columns, so it wasn't great for word processors or spreadsheets. People would use it as a home computer but didn't see it as a business computer at all. That was the PET line.[45]

Counter to this, Gail Wellington, head of Software Operations at Commodore UK, has stated that, because the VIC-20 was seen as a game computer, she wanted to focus on productivity software for the C64, to avoid competition with the older computer.[46] She also negotiated with Microsoft to publish their MultiPlan spreadsheet for the C64.[47] Certainly, Commodore sold a considerable range of serious C64 software for home and business use. The official 1982 or 1983 *Commodore 64 Software* catalog lists a dizzying array of "Home & Business Aids": EasyCalc 64, EasyFile 64, EasyFinance 64, EasyInventory 64, EasyMail 64, EasyPlot 64, EasySchedule 64, EasyScript 64, EasySpell 64, Future Finance 64, The Word Machine / The Name Machine (a simple word processor and address label generator), and Science Calculator 64.[48]

Early serious C64 software development was a gold rush akin to the rush to make arcade games, discussed in chapter 2. Figure 1.6 shows Microsoft's MultiPlan spreadsheet (left), which was only one of many spreadsheets. A 1984 *Ahoy!* article reviewed spreadsheets Bcalc, Busicalc 2, Calc Result Easy, Calc Result Advanced, ESP > Calc, MultiPlan, Omnicalc, Practicalc, and Spreadsheet Assistant, eschewing Commodore's EasyCalc.[49] This list does not even include German publisher Data Becker's Kalkumat. In fact, the German market for serious C64 software was by itself of sufficient size to support several large companies. In 1984, Data Becker was the largest personal computer software publisher in Germany, with a turnover of 40 million marks, publishing many German-language productivity programs.[50]

The same picture applies to word processors, including Commodore's EasyScript, VizaWrite, Data Becker's Textomat, and many more. Magazines would sometimes make small versions of such software available, like *Compute!'s Gazette*'s 1984 type-in word processor program SpeedScript (figure 1.6, right).[51] Word processors were at the time sufficiently new that the editor wrote a remarkably detailed introduction to how and why one would concretely use a word processor, how it compared to typewriters, and how it answered the question on everybody's mind—why do you need a computer?

> Documents are typed not on paper, but on the video screen. Characters, words, phrases, sentences, paragraphs, or even large blocks of text can be modified, deleted, inserted, added to, moved, duplicated, and manipulated to your fingers' content. When everything is just right, you can print out as many perfect copies as you want. . . . When I hear the cliché that computers (particularly home computers) are "a solution in search of a problem," I argue that even if computers were good for nothing else but word processing, it would be enough to justify their existence.[52]

► Try SpeedScript word processor on the website.

Starting to Program the C64

You've opened the box; you've connected the computer to the tape or disk drive and television. The *Commodore 64 User's Guide* (sometimes *User's Manual*), on the right side of the table in figure 0.1, was distributed with the machine and explains how to set up the C64 and how to program in BASIC.[53] We can use this to gauge the intentions of at least part of the Commodore company at the launch of the C64. On the very first page of the introduction, half a page is spent with an overview, and the guide then dives into the option of using sprite graphics for creating your very own "arcade-type video games." The guide writers expected game playing to be central for consumers, *who would be trying to make games themselves*.

Chapter 1 of the *User's Guide* explains how to set up the computer, chapter 2 introduces the keyboard, loading and saving of programs, and how to use the computer as a simple calculator by typing in BASIC commands directly.

```
PRINT 12 * 12
144
```

It is a rule in life that early feedback is better than late feedback, and this explains some of the popularity of BASIC on the C64. (► Try typing BASIC commands on the website.)

Programming BASIC

BASIC is a simple and forgiving programming language that, unlike more “professional” languages, does little to enforce programming structure or conventions.

On the C64, you can write BASIC commands on their own when the computer has started, but a BASIC program is structured around line numbers, like the program for printing out your name, where line 10 prints the text to the screen, and line 20 instructs the computer to “go to” line 10.

```
10 PRINT “COMMODORE 64!”
20 GOTO 10
```

Variables (used to remember things) are by default floating point numbers, but a “string” (a piece of text) is marked with $.

```
10 INPUT “WHAT IS YOUR NAME”;A$
20 PRINT “WELCOME, “+A$
```

To do something 10 times, such as counting from 1 to 10, a FOR/NEXT “loop” is used:

```
10 FOR I=1 TO 10
20 PRINT I
30 NEXT I
```

To use many of the C64’s advanced graphics and sound capabilities, we must use PEEK (reading a memory location) and POKE (changing a memory location) commands. For example, this one-line program changes the screen color based on the key being pressed:

```
10 POKE 53281,PEEK(197):GOTO 10
```

In the third chapter, the *User's Guide* introduces the common simple program of printing out "Commodore 64" across the screen, but realistically most users changed it to their own name. (See the "Programming BASIC" box.) In the introductory chapter example (figure 0.2), I added the C64-specific command POKE 646,RND(1)*16, which randomly changes the text color every line. Such programs were a staple of computer shops and classrooms. It was also many people's first experience of the power of programming, where you literally made the computer do work for you—here the noble work of filling the screen with your name. The guide proceeds to teach users about variables, IF/THEN and FOR/NEXT loops.

It is only in chapter 4 that the manual moves beyond the teletype terminal tradition to create a ball that bounces from left to right. This is followed by a Fahrenheit/Celsius converter demonstrating how to get input from the user, a lesson in random numbers, and finally the first game, GUESSING GAME, which combines the previous lessons, asking the player to guess a random number, clearly based on the *Guess* game from Ahl's collection.

This segues into the RANDOM GRAPHICS example (figure 1.7) that generates a maze by randomly switching between two diagonal lines. This example has since served as the focus of an entire book, *10 PRINT CHR$(205.5+RND(1)); : GOTO 10*.[54]

Chapter 5 of the manual shows how to PRINT colors, and how to use PEEK and POKE commands to read and change memory locations in the C64 directly. PEEK and POKE were introduced into the BASIC language by Bill Gates and Paul Allen, giving users the ability to directly control the machine in ways that would not work in a time-sharing system like Dartmouth's, due to the power it gives the user to crash the system.[55] To illustrate, the following command:

```
POKE 1,0
```

immediately locks up the C64, forcing the user to turn it off and on again. PEEK and POKE would thus have posed a problem on time-sharing systems like Dartmouth's, used by many users simultaneously. Is it not a bug that the user can crash the C64 so easily? It helps to understand that while modern computers and devices are meant to never crash, no matter what the user does, 1980s machines delegated much of this responsibility to

users and programmers. It is like the old joke about a patient visiting their doctor, contorting their arm to an absurd position, and complaining, “It hurts when I do this!” To which the doctor replies, “Then stop doing it.” If you did something to make the computer crash—then you should just stop doing it.

Figure 1.7 RANDOM GRAPHICS program example from the *User’s Guide* generating a maze by randomly printing left- and right-tilting lines.

On the C64, PEEK and POKE served as the only way for users to use most of its graphics and sound abilities. Even changing the background color to black could only be achieved with a POKE, with 53281 representing the memory location of the background color register in the VIC-II chip and 0 representing the color black:

```
POKE 53281,0
```

All C64 audiovisuals beyond colored text and PETSCII must be created through directly changing the C64’s memory locations while avoiding crashing the computer by changing the wrong location. This system certainly leaves much to be desired, as C64 reviews would criticize, requiring users to memorize the memory locations for manipulating the VIC-II graphics and SID sound chips.[56]

At this point, the *User's Guide* explains the memory layout of the machine and how to directly manipulate screen memory to make a ball bounce around the screen. My personal recollection is that when I tried making this program, I often made mistakes and my program POKE'd—and garbled—the program in memory rather than the screen, but the manual had not prepared me for the possibility. I was genuinely worried that my C64 was broken and raised my concern with a nonplussed computer store employee.

In chapter 6 of the manual, we finally get to the sprites promised in the marketing material. Figure 1.8 shows the limited graphical tools of the time: users are encouraged to draw their sprites on graph paper and do the binary arithmetic need to convert their designs into the data statements depicted in the figure. Once you get the hang of it, making a sprite and displaying it on screen is straightforward, and tools also became better over time. Sprites are also much easier to use than any other way of displaying moving graphics, and fast enough that you can use them for making some games in BASIC.

Chapter 7 on "Creating Sound" is the last chapter to deal with C64-specific abilities. It explains the structure of the C64 chip and provides a "Michael, Row the Boat Ashore" sample program, as well as two examples of sound effects, a "DOLL CRYING" and "SHOOTING SOUND." The final chapter is about "Advanced Data Handling," offering more traditional programming discussions of data structures in BASIC. (► Try BASIC programs on the website.)

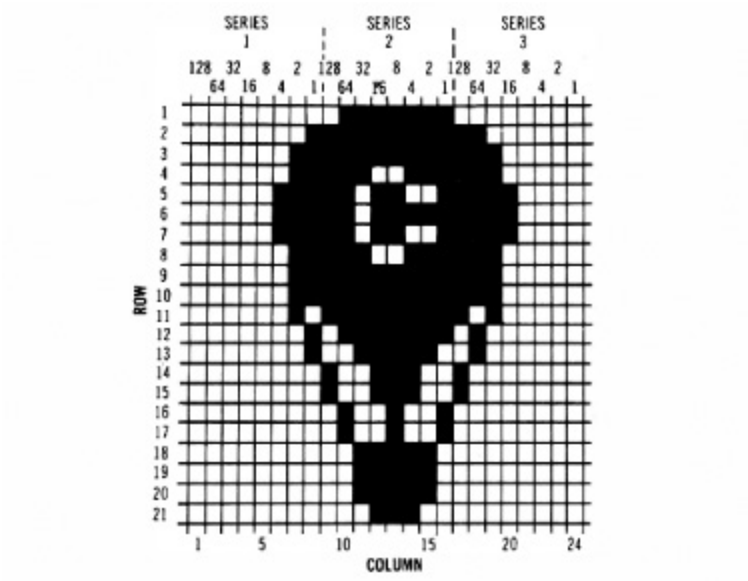
A Culture of BASIC

What did people use C64 BASIC for at the time? Judging by magazines, typing in programs and programming in BASIC were popular activities. Over time, type-in programs became increasingly complex. In Germany, a popular type-in program was *Bundesliga Manager*, an involved simulation of the 1985–1986 German football league season. Going back to the relevant *Happy Computer* magazine reveals eight and a half pages of tightly spaced program.[57]

Even without using sprites, it was possible to create some action games (those with mostly static graphics) in BASIC on the C64. Figure 1.9 shows my 1985 version of *Snake*, the display constructed of PETSCII characters.

Because the game only updates a small portion of the screen—the tip of the snake—BASIC is fast enough. Similarly, it was easy to implement the *Tron Light Cycles* game, where two players must surround each other. This game, a two-player version of *Snake*, was known from the 1982 *Tron* movie and arcade game.

As BASIC was implemented on a range of computers, not everyone adhered to the original BASIC design, and manufacturers added relevant commands for showing graphics or playing sound, gradually making BASIC dialects less compatible with each other. The language originators, who saw themselves as designing a democratic language for all to use, disparagingly referred to these new variants as "street BASIC," "a horrible dialect of a beautiful language."[58] The 1982 Dutch BASICODE project tried to solve the issue of incompatible BASIC dialects with a shared set of libraries for creating compatible BASIC programs across a range of machines, as well as for transmitting programs on the radio in a way that would be readable on cassette tapes for the same range of machines.[59] Ultimately, though, BASIC dialects remained incompatible.



Now look at the design of the balloon. The first series of 8 squares on the first row are all blank, therefore the bits are all OFF giving a value of zero. The middle series on row 1 looks like this:

00000000 01111111 00000000

The third series of 8 squares on the first row also contains only blanks so it also equals zero. So the data for the first line is:

DATA 0, 127, 0

The three series of dots that make up row two are calculated like this:

Series 1: 0 0 0 0 0 0 0 1

1 = 1

Series 2: 1 1 1 1 1 1 1 1

128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 255

Series 3: 1 1 0 0 0 0 0 0

128 + 64 = 192

Figure 1.8 The *Commodore 64 User's Guide* explains how to use sprites to design your own games by drawing on graph paper and doing the binary math needed to convert the pixels into programming.

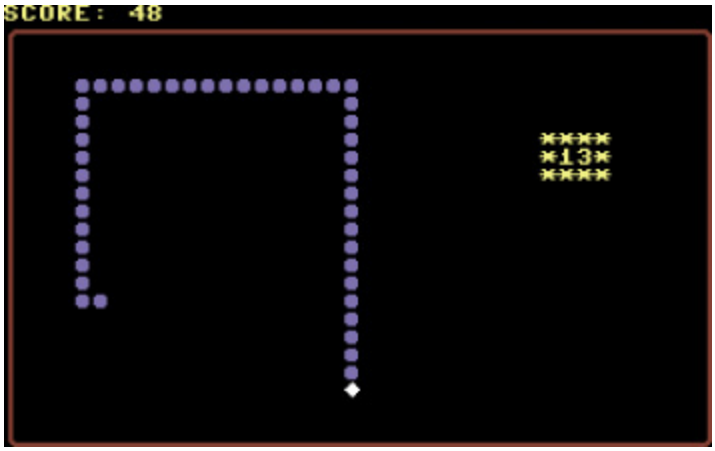


Figure 1.9 The author's 1985 BASIC version of *Snake*, easily programmed on the C64 using the PETSCII character set. Just as the BASIC tradition influenced C64 homebrew development, so too did games from other platforms.

C64 BASIC connected the C64 to a tradition of BASIC programming, but it suffered the problem that C64 graphics and sound features were poorly supported and that the kind of games the C64 was promoted on could not be made in BASIC. This pushed users away from BASIC, disconnecting the C64 from the BASIC tradition, making machine language an enticing next step for users.

2

The Arcade Game Machine (and Beyond)

Imaginary #2:	The Commodore 64 brings arcade games home!
	The Commodore 64 takes video games beyond the arcade!
Programming Language:	Machine language
Graphics:	Custom character sets, with sprites on top
Colors:	Colored objects on a black background; later embossing
Quotes:	At first it seemed that Commodore wished to promote the new system as a business machine . . . beneath that unpossessing beige exterior beat a heart of pure gaming, with a comfy and familiar 6510 CPU harnessed to arcade-quality sprite and character map displays and an extraordinarily versatile and excellent-sounding synthesizer chip —the SID—which became legendary in its own right. —*Jeff Minter,* The Commodore 64 Book*, 2007*

	By far and away the most common use of a 64 is running the superb games and entertainment software available for it. —Zzap!64, *May 1985*
	BASIC is just too slow. Anyone who's programmed on the Commodore 64 knows that for arcade game speed, or for simulation sophistication (and its accompanying complexity), nothing beats machine language. —Compute's Machine Language Games for the Commodore 64, *1986*
	REMEMBER ITS NOT JUST A PLATFORM GAME ITS AN ADVENTURE. —Wanted: Monty Mole, *1984*
Examples:	jesperjuul.net/c64/2/

Arcade Envy

The Commodore 64 launch was followed by a deluge of arcade games. At this time, physical arcade games were central in video game culture, and it was an essential attraction of home systems that you could bring arcade games home or, even better, program them yourself: renowned UK game developers the Oliver Twins have told the story of playing arcade games as teenagers and deciding that, "what we really wanted to do was to write proper arcade action-style games."[1]

Consider the four 1983 and 1984 games from UK company Interceptor Software shown in figure 2.1. In these early days of home video game development, copyright was not consistently observed, and Interceptor published a series of blatant clones of arcade games, including *Frogger 64* (*Frogger*), *Spriteman 64* (*Pac-Man*), *Siren City* (*Rally-X*), *Crazy Kong 64* (*Donkey Kong*), *Cuddly Cuburt* (*Q*Bert*), *Scramble 64* (*Scramble*), *Defender 64* (*Defender*), and *Missile Command* (well, *Missile Command*). The games were made in a rush, made poor use of the C64's capabilities, and a single Interceptor developer, Ian Gray, published no fewer than eight games in 1983. Better versions, often licensed, arrived gradually. This abundance of unlicensed clones was not unique to the C64—for example, the Dragon 32 computer had the equally unlicensed *Donkey King*.[2] The standout game in the above list is Gray's *Siren City*, based on *Rally-X* but set in a crime-ridden city that appears to have inspired the *Grand Theft Auto* games, a connection I will return to at the end of the chapter.

We can admire the exploitative audacity of the developers while questioning their legal wisdom, but video game copyright was neither well established nor consistently enforced at the time. In a 1983 interview with 17-year-old "whizz kid" Richard Jones, proprietor of Interceptor, Jones explained how certain games could only be marketed in the UK, "because you cannot sell them in the US where copyright rules are stricter. . . . When I showed Frogger 64 at a computer show in America, I was told to turn it off immediately. The dealers were afraid of a law suit if they showed it for even just a few minutes."[3] While it was possible to sell an unlicensed clone of *Frogger*—also titled *Frogger*—in some European countries, there was a US version of *Frogger* properly licensed by Sierra/Parker Brothers from Sega, and legal precedent became settled in the US before other countries.[4]

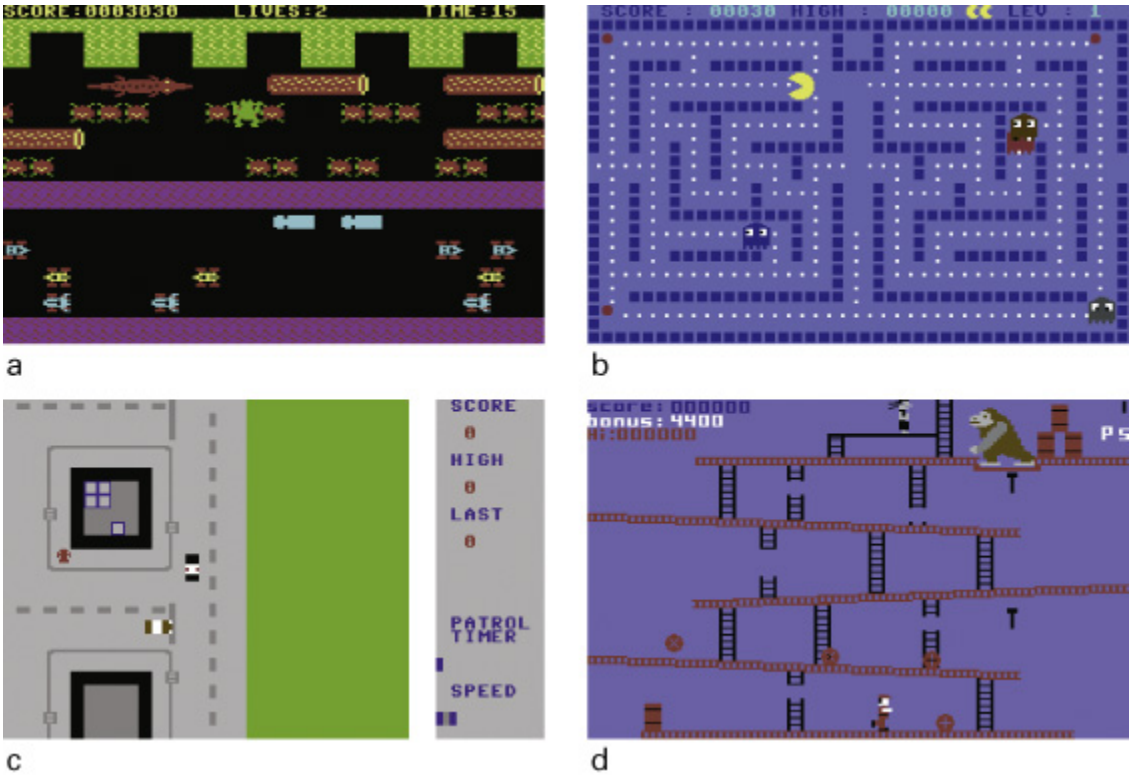


Figure 2.1 The hope of bringing arcade games home was central in early C64 development. A small sample of the unlicensed arcade-inspired games published by UK Interceptor Software in 1983 and 1984: a) *Frogger 64*, b) *Spriteman 64*, c) *Siren City*, and d) *Crazy Kong 64*.

1980s home computers and home consoles eventually contributed to a historical conundrum: they ended up nearly killing the very arcade game industry that they had been designed to emulate. Developers were, however, faced with other challenges. Developers had to adapt to different playing situations (at home with unlimited time) and business models (upfront purchase for a game), and many game types appeared in response to this new situation. Developers also had to make machines designed for older game types run new games.

In this chapter, I show how the C64 gradually came to be seen as a game computer—in a positive sense—even though Commodore had tried to avoid that label. I uncover new insights into the relationship between US and European markets and into the limited role of consoles in Europe in the 1980s. I then show how the C64 hardware was used to develop common arcade games and later a C64-specific arcade-adventure tradition. This chapter is a study of how developers used the C64 hardware in both obvious

and original ways, finding ways to move beyond arcade game conventions. I look at music on the C64, the role of piracy in C64 culture, and finally I present an overview of the role of the C64 in video game history.

A Game Machine and Proud

As shown in chapter 1, Commodore knew game-playing to be a controversial use of the C64 and tried to make it easier for a potential buyer —like a parent—to rationalize that they were buying a serious machine. The fear of games was pervasive in advertising, hardware design, and corporate communication. In 1983, as home computers were becoming popular in Germany, a *Der Spiegel* article concluded that home computers were often used for playing games, and the managing director of Apple Germany complained that games were ruining the reputation of computers.[5] In 1985, Bruce Webster argued that Apple should never make a Macintosh with color abilities, and I referred to his case that IBM refrained from adding game-related graphical abilities to the IBM PC because they were afraid of the stigma of game machines.[6]

Rather than fear them, some chose to fully embrace the C64 and games. In the inaugural 1985 issue of UK *Zzap!64* magazine (figure 2.2a), editor Chris Anderson explained the need for a magazine focused solely on the C64, and only on games:

> Until now Commodore 64 fun-lovers in search of a good read have had to choose between the general games magazines (and put up with boring stuff about Spectrums, etc) and the oh-so-serious Commodore magazines (and put up with pages of complex computer jargon written by unintelligible boffins). Which is strange, because by far and away the most common use of a 64 is running the superb games and entertainment software available for it.[7]

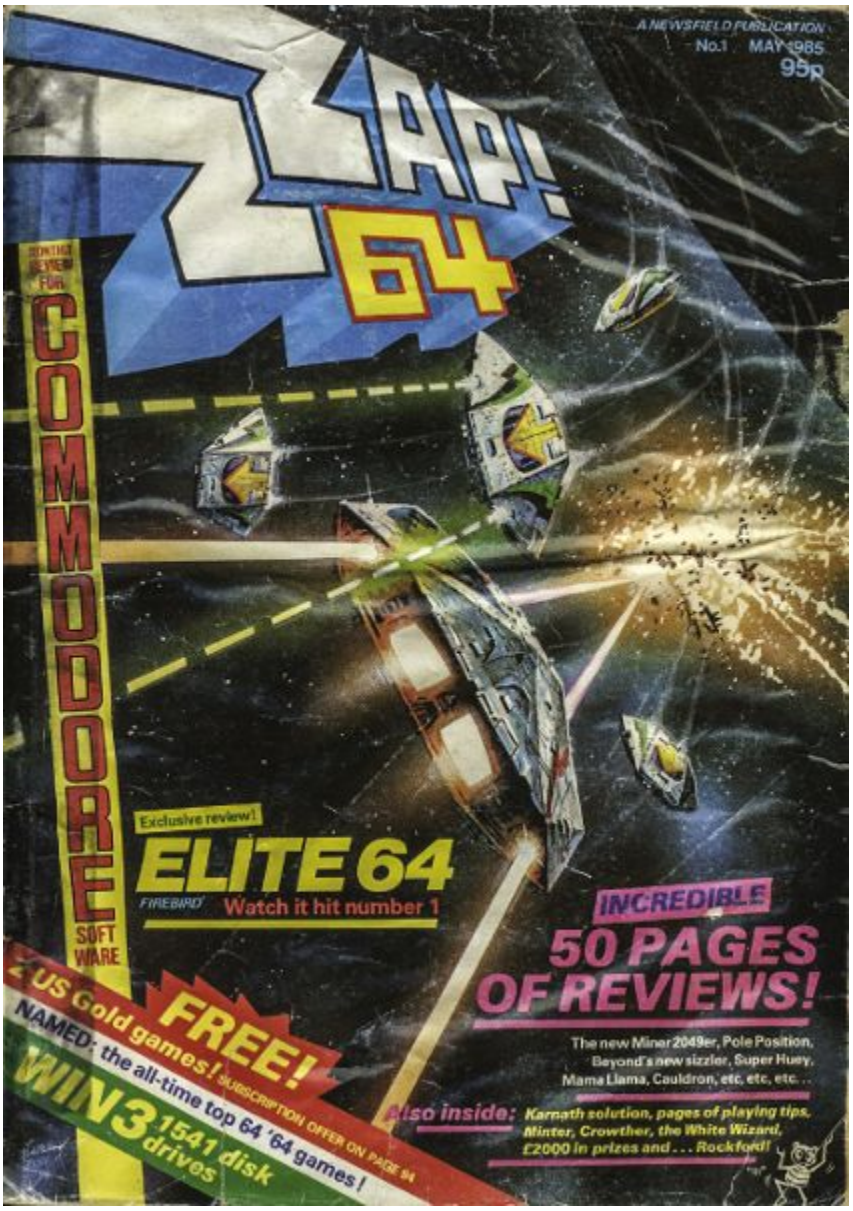


Figure 2.2a Inaugural issue of the UK game-oriented *Zzap!64* magazine

Figure 2.2b US *Compute!'s Gazette* general Commodore computing magazine

Figure 2.2c The more technical German *64'er Magazin*.

Zzap!64 promised readers that they would not find the "type-in program listings" covering sometimes dozens of pages. *Zzap!64* thereby marked a shift to seeing readers as consumers of packaged software.

We can use C64 magazines of the 1980s and early 1990s to understand how C64 culture developed over time and begin to understand the relationship between game and computer culture. Throughout this chapter, we will look primarily at three magazines: UK games magazine *Zzap!64* (1985–1992/1994; figure 2.2a), the US general Commodore magazine

Compute!'s Gazette (1983–1994; figure 2.2b), and the German general C64 magazine *64'er Magazin* (1984–1994; figure 2.2c). I have selected these because they represent large territories of the C64, and because they approach the C64 from different angles and represent different commercial trajectories of the platform. I have also sampled other magazines like Swedish *Datormagasin* (1986–1995), US *Ahoy!* (1984–1989), UK *Commodore Format* (1990–1995), and German *Aktueller Software Markt* (*ASM*) (1986–1995).

Why would *Zzap!64* want to discuss video games, which are obviously software programs, without discussing programming at all? In his influential study of early UK video game magazines, *The Formation of Gaming Culture*, Graeme Kirkpatrick identified an important shift in the mid-1980s: in early game and computer magazines, video games were grouped with other software, and reviews would often refer directly to programming.[8] But as the 1980s went on, games became separated from software, and game reviews ceased to employ technical language.[9] To explain this, Kirkpatrick employs sociologist Pierre Bourdieu's concept of a *field*. Similar to the way the art world has its own codes and conventions, participants in game culture also act in certain ways, have their own ideas of what is considered important, and have their own codes to show that they belong to the culture.[10] As Kirkpatrick describes it, "After 1985 . . . gameplay is opposed to things like graphics, character, plot and so on. . . . The true gamer . . . is the player who is interested in games and not computers; gameplay and not stories, and graphics in so far as they reward good play."[11]

The word "gameplay" became central in the language of game culture, and to use it was to show that you belonged to this culture. Game culture—according to Kirkpatrick—was therefore formed by rejecting technical topics and thereby becoming distinct from the culture of computing and programming. This is a strikingly precise explanation of *Zzap!64*'s rejection of technical jargon. (Chapter 3 shows that counter to that initial rejection, the magazine began to write about programming the following year.)

Regardless of Commodore's advertising efforts, and as other computers took the mantle of "seriousness," the dominant imaginary of the C64 switched to being that of a game machine. In Europe, Commodore gradually warmed to marketing the C64 as a machine for games. In 1989,

the *Batman* bundle packed a C64, a Datasette, a joystick, the *Batman* game, and some additional games into a black box with the *Batman* logo. Other bundles included the 1990 association football *World Cup* bundle (coinciding with the 1990 World Cup in Italy). Then-UK national accounts manager David Pleasance described this as, "From this point on we do not sell computers—we sell dreams."[12]

The History of C64 Games and the US–European split

The C64 was a platform for commercial video games from 1982 to the early 1990s. Figure 2.3 shows the yearly number of game releases for the C64 according to the MobyGames database.[13] The database is maintained by volunteers and is likely to favor fondly remembered games. Still, it gives us an outline of the trajectory of the C64 as a game platform. We see a peak in 1984, followed by a 1985–1986 lull, a new peak in 1987, and a gradual decline until the early 1990s. Reading Commodore's financial statements in 1986, we can see a dip in the US market, while the non-US market (focused on Europe) is twice as large, representing how the center of gravity for C64 culture shifted to Europe during the late 1980s.[14]

The US version of video game history typically features a pivotal "video game crash" in 1982, eventually leading to Atari burying its subpar *E.T.* game in the New Mexico desert.[15] However, this crash did not manifest much outside the US and was mostly confined to game consoles. At the time of the crash, an analyst saw the computer market as distinct from the game market and "did not expect similar troubles in the related $6 billion home computer market."[16] Commodore developer Andy Finkel told me that the crash had no influence on video game development at Commodore.[17] Developer Chris Crawford claims a central feature of the crash was that "home computers replaced home video games [consoles]."[18]

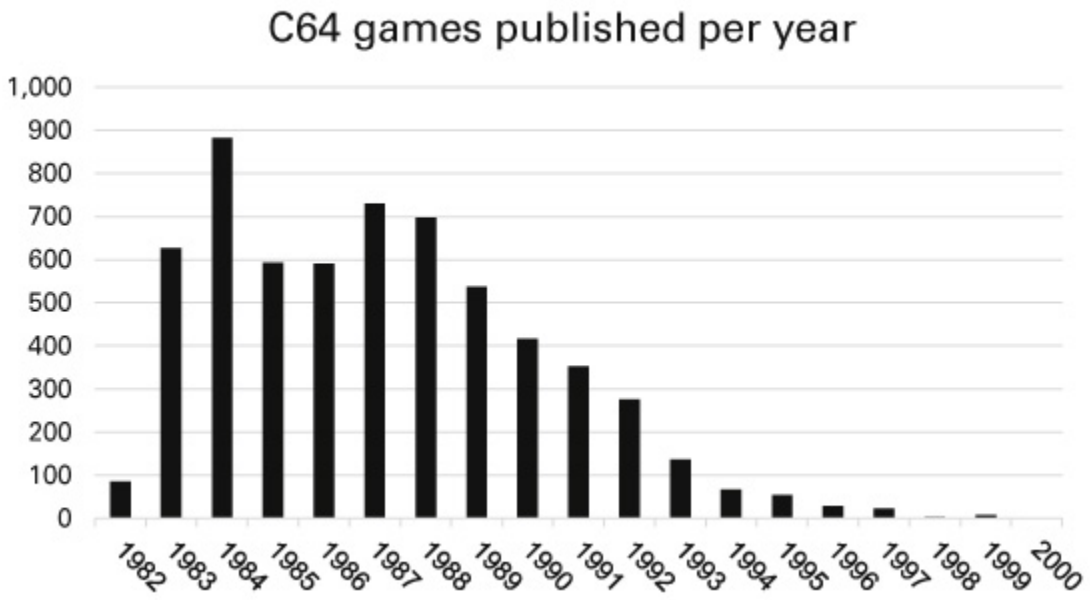


Figure 2.3 C64 games published per year (according to the MobyGames database), showing the C64's early peak in 1984 and gradual decline as commercial platform from 1989 to 1993.

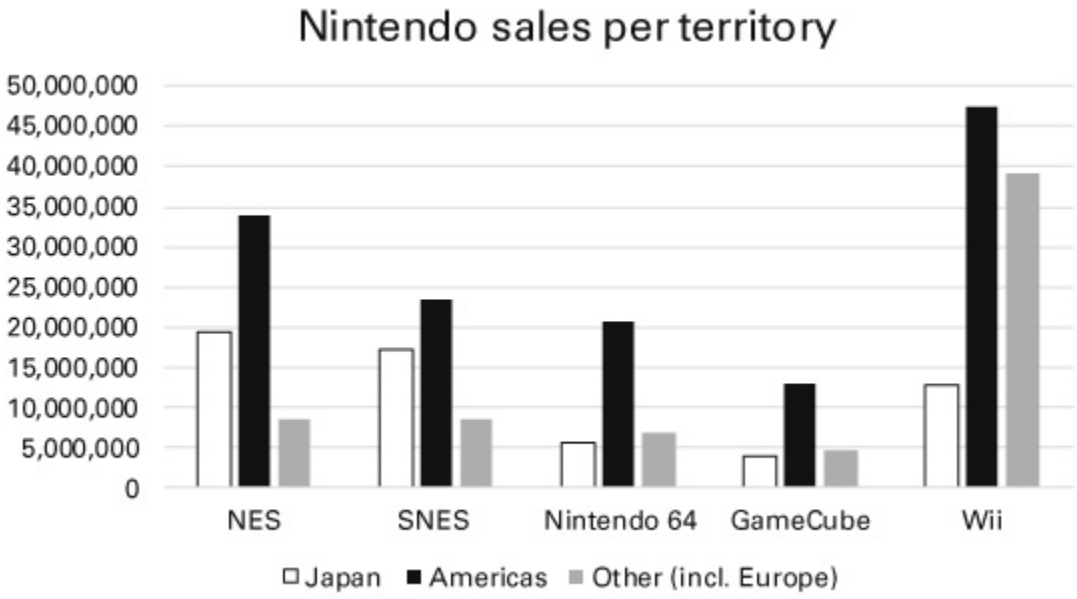


Figure 2.4 Nintendo sales by console and region, showing that early consoles, including the Nintendo Entertainment System, were much less popular in Europe/Oceania than in the US.

Many video game histories place considerable emphasis on Nintendo, and it can therefore be helpful to understand the relative strength of Nintendo in different territories. Visualizing Nintendo's own sales figures (figure 2.4), we see that sales of both the Nintendo Entertainment System (NES, launched 1986–1987 in Europe) and the Super Nintendo Entertainment System (SNES, 1990–1992) were modest in Europe compared to Japan and the Americas.[19] (Europe was by far the largest market in the "other" category, and North America was the largest market in the Americas.) Also note that in Europe, the NES was launched four to five years after the C64 and ZX Spectrum, which were by then entrenched as the most popular video game devices. In Europe, it was not until the Wii (2006) that Nintendo achieved the kind of market penetration the NES had had in the US. This explains why Nintendo is often synonymous with early video game history in the US and Japan but is less so in Europe.[20] In Europe, the Sega Master System was launched in 1987 and came to be the more successful home console in the region.[21] Still, video game console software sales did not become significant in the UK until 1991.[22]

A Tale of Two C64 Markets

But what was the relationship between the US and the European market? From 1985 to 1992, *Zzap!64* featured a readers' chart (figure 2.20) where readers would write in and vote for games by mail (replaced by a Gallup sales chart in 1993). The readers' chart is interesting because it is the longest-running C64 games chart available. We examined the top ten games of all *Zzap!64* charts and counted the number of weeks a game spent on the charts. Figure 2.5 shows all games spending at least four months on the charts. This does not tell us which games sold the most, but given the prevalence of piracy on the C64, to be discussed later, readers' charts are likely better indicators than sales numbers of what games were played.

Where were the games in figure 2.5 from? We examined the country of origin of all the games on the charts, plotting them as shown in figure 2.6. We designated the country of origin as the *original* origin of a game and analyzed games according to the year of release. A C64 conversion of a Japanese game from Taito is thus counted as Japanese regardless of where the conversion was made. We can see that the early years were dominated

by US games, but that those were gradually replaced by UK games from 1984 to 1987. From 1988, we see an increasing percentage of Japanese games, mostly arcade conversions like *Bubble Bobble* (1987) and *Rainbow Islands* (1990)—conversions made by UK companies—as well as a small share of Hungarian, Dutch, and German games. The C64's arcade games were initially unlicensed clones, but toward the end of its commercial life, the C64 had official arcade conversions as companies like UK developer/publisher Ocean Software cultivated connections to the Japanese arcade industry.[23] The big story here is the disappearance of US software from this UK chart, showing that even as Commodore was a dominant home computer manufacturer in the US in 1986, the global reach of the US C64 software industry was waning.

The difference between US and European markets is also visible in the software charts of the time. The November 1985 *Popular Computing Weekly* UK game chart in figure 2.7 shows that the ZX Spectrum and C64 were the preeminent game platforms in the UK, and that nearly all of the top twenty games were made for both machines.[24] The third major platform on the list is the Amstrad CPC. A German *Aktueller Software Markt* chart around the time shows a similar platform distribution.[25]

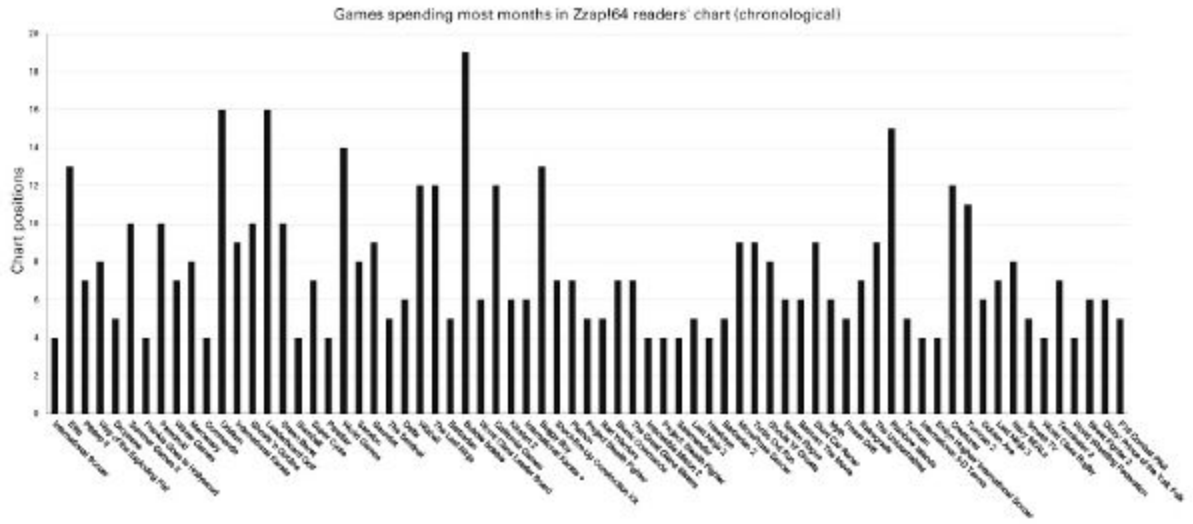


Figure 2.5 Games spending at least four months on *Zzap!64* readers' chart, showing the most popular C64 games in the UK by vote from 1985–1992 (listed in chronological order).

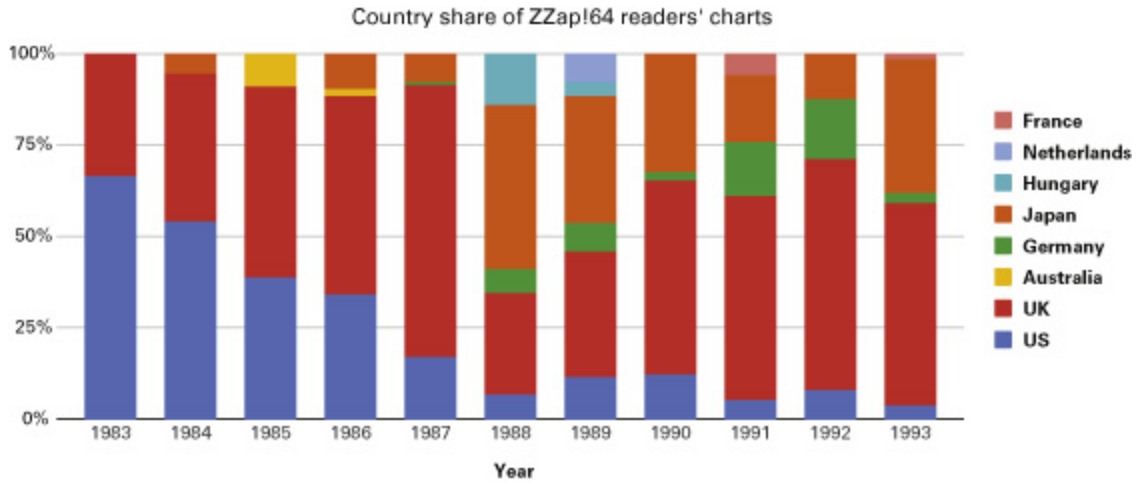


Figure 2.6 Country of origin (original game) of *Zzap!64* charts, showing a shift from US software to games with UK, German, and Japanese origins (listed by release date).

The *Compute!* US charts of the same month show an entirely different computer landscape, with the Apple II, C64, and Atari 8-bit being the main platforms, followed by the IBM PC and Macintosh.[26] The following year, 1986, Atari 8-bit computers were gradually dropped by US developers, and the *Compute!* charts show games released primarily for the C64, Apple II, and then IBM.[27]

The C64 was the only major platform on all 1985 charts, the only transatlantic computer. After the waning of Atari's 8-bit computers as a commercial platform, the C64 was the only popular platform (until Commodore's own Amiga) designed for fast-moving arcade games. This created a duality where, on one hand, it was expedient for publishers to envision cross-platform games that would work on different machines, but on the other hand, reviewers and players often celebrated games that exploited the C64's specific abilities. I discuss the resulting C64-specific tradition of action-adventure and scrolling games later in this chapter.

Besides the different platforms, there were two other standout differences between US and European C64 software. First, the US software industry was older than the European industry. For example, US developer/publishers Sierra Online and Activision were founded in 1979 for, respectively, Apple II and Atari 2600 games, and US Epyx Games was founded in 1978 for Commodore PET games. On the other side of the Atlantic, however, UK Ocean and German Data Becker were both founded as publisher/developers in 1983, and German *Turrican* developer Rainbow Arts was founded in 1984.[28] The US already had companies and distribution methods organized by the time the C64 launched, whereas many European companies started in response to the popularity of the C64 and/or ZX Spectrum. The second difference concerns the kind of games made. Generalizing, popular US games were often heroic, offering the player either secret agent roles (*Impossible Mission*), Olympic medals (*Summer Games*), military victories (*Beach Head*), or common daydreams (*Sid Meier's Pirates!*). By comparison, especially early UK games were strange, full of antiheroes in surreal and mundane settings, from *Attack of the Mutant Camels* to *Frankie Goes to Hollywood* to *Monty on the Run*. It appears the US industry had already settled on well-known and Hollywood-compatible narrative tropes, while UK developers were happy with the

underdog role and were making games whose draw came in part from the experimental and surprising.

Top Twenty

1	(–)	Monty on the Run *(Spectrum/C64)*	**Gremlin Graphics**
2	(1)	Way of the Exploding Fist *(Spectrum/C64/Amstrad)*	**Melbourne House**
3	(3)	Frank Bruno's Boxing *(Spectrum/C64/Amstad)*	**Elite**
4	(4)	Impossible Mission *(Spectrum/C64)*	**Epyx/US Gold**
5	(7)	Formula One Simulator *(Spectrum/C64/C16)*	**Mastertronic**
6	(6)	Finders Keepers *(Spectrum/C64/Amstrad/MSX)*	**Mastertronic**
7	(8)	Daley Thompson's Supertest *(Spectrum)*	**Ocean**
8	(13)	Action Biker *(Spectrum/C64)*	**Mastertronic**
9	(–)	Starquake *(Spectrum)*	**Bubble Bus**
10	(4)	Hacker *(Spectrum/C64/Amstrad/Atari)*	**Activision**
11	(9)	Fighting Warrior *(Spectrum/C64)*	**Melbourne House**
12	(14)	BMX Racers *(Spectrum/C64/C16)*	**Mastertronic**
13	(6)	Who Dares Wins II *(C64)*	**Alligata**
14	(15)	Graham Gooch's Test Cricket *(Spectrum/C64)*	**Audiogenic**
15	(11)	Fairlight *(Spectrum)*	**The Edge**
16	(–)	Chiller *(Spectrum/C64/Amstrad/MSX)*	**Mastertronic**
17	(18)	Beach-Head *(Various)*	**Access/US Gold**
18	(19)	Bored Of The Rings *(Spectrum/C64/BBC)*	**Silversoft**
19	(–)	Nonterraqueous *(Spectrum/Amstrad)*	**Mastertronic**
20	(5)	Now Games *(Spectrum/C64)*	**Virgin**

Figures compiled by Gallup/Leisurescope

HOTWARE: Software Best Sellers

This Month	Last Month	Title	Publisher	Remarks	Systems: Apple	Atari	Commodore	IBM	Macintosh
Entertainment									
1.	2.	*F-15 Strike Eagle*	MicroProse	Air combat simulation	•	•	•	•	
2.	3.	*Flight Simulator II*	SubLogic	Aircraft simulation	•	•	•		
3.		*Gato*	Spectrum Holobyte	Submarine simulation	•			•	
4.	5.	*Karateka*	Brederbund	Action karate game	•		•		
5.	4.	*Flight Simulator*	Microsoft	Aircraft simulation				•	
Education									
1.	2.	*Typing Tutor III*	Simon & Schuster	Typing instruction program	•		•	•	•
2.	3.	*Math Blaster!*	Davidson	Introductory math program, ages 6-12	•	•	•	•	
3.	1.	*New Improved MasterType*	Scarborough	Typing instruction program	•	•	•	•	•
4.		*Charlie Brown's ABC's*	Random House	Letter and word introduction, ages 3-4	•	•		•	
5.	5.	*Music Construction Set*	Electronic Arts	Music composition program	•	•	•		
Home Management									
1.	1.	*Print Shop*	Brederbund	Do-it-yourself print shop	•	•	•		
2.	4.	*The Newsroom*	Springboard	Do-it-yourself newspaper	•	•	•	•	
3.	2.	*Print Shop Graphics Library*	Brederbund	100 additional graphics	•	•	•		
4.		*Print Master*	Unison World	All-home print shop				•	
5.		*Bank Street Writer*	Brederbund	Word processor	•	•	•	•	

Figure 2.7 Best-selling games in November 1985 according to *Popular Computing Weekly* in the UK (top) and *Compute!* in the US (bottom). Not only are the games different, so are the platforms. Main UK platforms are the ZX Spectrum, C64, and Amstrad CPC. Main US platforms are the Apple II, Atari 8-bit, C64, and IBM PC. The C64 is the only transatlantic platform.

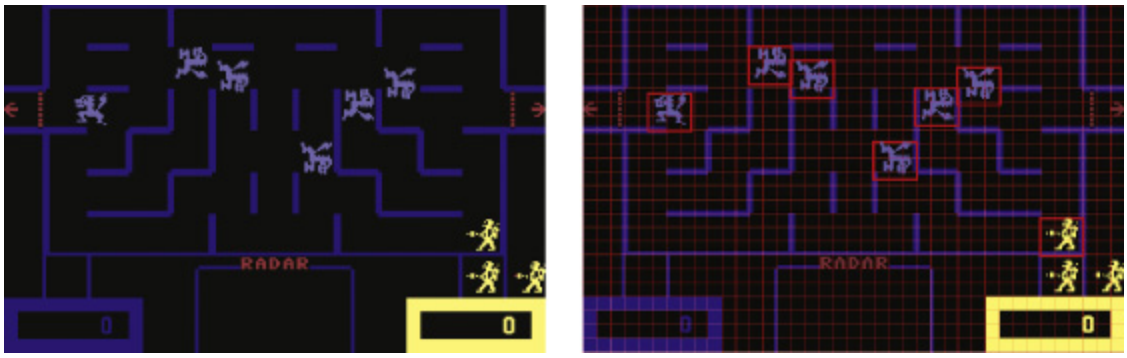


Figure 2.8 *Wizard of Wor* (Commodore, 1983) arcade game license. Left: Game screen. Right: Sprites marked with red outline and characters with a grid.

Wizard of Wor: The Arcade Comes Home

How did the arcade fascination of C64 designers translate into games in practice, and how could developers use the C64 to move beyond the arcade tradition? Let us first look at Commodore's 1983 C64 version of *Wizard of Wor*, converted from a 1980 arcade game by Bally/Midway, interesting for both its technical details and for the broader context around it. Figure 2.8 shows the game and a second image marking the character layer in a grid and the sprite elements with red outlines. Developer Jeff Bruette defined his own character set and sprites (figure 2.9), including an enlarged alphabet and numbers, as well as some vertical and horizontal lines for the maze. In the screenshot, the yellow astronaut controlled by the player is a sprite, but the two visually identical astronauts representing extra lives are made from multiple characters—in the character set, we can see a set of feet just above A and B. Because the game could have six enemies and two moving astronauts, Bruette had to work around the C64's eight-sprite limitation by creating the static astronauts representing lives left from characters.[29] Shots, made in characters, are the only moving objects that are not sprites. Level start is signaled with a simple minor, foreboding musical figure (D, E, F, D). During levels, music consists of a descending three-note pattern that speeds up. Shots and explosions are made using the noise generator of the SID chip. *This is exactly what the C64 was designed for*. (► See *Wizard of Wor* video on the website.)

Figure 2.9 *Wizard of Wor* characters and sprites. Top: Custom-defined character set featuring regular letters, larger numbers combined from several characters, and characters for the maze, shots, and astronaut. Bottom: Sprites featuring animations of player and enemies.

We can also consider the game in the larger physical context of the C64. *Wizard of Wor* comes on a cartridge, it is a two-player game, matching the two joystick inputs on the right side of the C64. Furthermore, it supports Commodore's short-lived *Magic Voice* device, providing then-technologically amazing speech, such as "Get Ready, Warrior!" at game start. For development, Bruette used the PET-family Commodore 8032 computer with dual disk drives, transferring the program to a C64 using a cable.[30]

Yet why did Commodore publish *Wizard of Wor*, of all the arcade games out there? Because, in 1981, Commodore had negotiated a deal with Bally/Midway, licensing several of its games including *Wizard of Wor*.[31] Andy Finkel explained to me the process of selecting games from the Bally/Midway catalog every year and developing games without access to the source code. Finkel felt this was an incidental job benefit: "We had a deal with Bally saying we could get perhaps six games a year of their new line. We'd decide which ones we thought would sell, and which ones the 64 could do, and they would send us the machines. They wouldn't give us the source code. They wouldn't even tell us the rules of a game or the scoring, so we just had to play it for hours and hours and hours. It was rough."[32]

How to Make Arcade Games on the C64

Even with sprites, C64 BASIC is too slow for making most arcade games including *Wizard of Wor*, in part because BASIC is centered on floating point numbers which are inefficient due to the rudimentary math abilities of

the 6510 CPU. How can we move from BASIC to machine language (understood here to be both the native language that the CPU understands and its human-readable form known as assembly language)? In BASIC we use the following POKE command to set the border color to black:

```
POKE 53280,0
```

In 6510/6502 machine language (see “Programming 6502 Machine Language” box), changing a memory value requires two commands, one to load the value 0 into the **A** register, and one to store the value of **A** in the memory location. (Semicolons represent comments to make the program readable by humans.)

`LDA #0`	`;LoaD 0 into A`
`STA 53280`	`;STore A in 53280`[33]

What a CPU mostly does is process and move data around in memory, but fortunately C64 graphics and sound are conveniently controlled by giving the VIC-II and SID chips instructions through changing memory addresses.

When first programming in machine language on the C64, the experience is one of staggering speed. A small BASIC program for filling the screen with the heart character takes 5.3 seconds, slowly working while we watch, but a machine language program doing the same takes 7 milliseconds, making machine language 750 times faster than BASIC, the hearts appearing instantly to our eye.[34]

Programming 6502 Machine Language

The 6510 CPU takes care of all logic and calculation in a C64 program. The 6502 family, of which the 6510 is part, has three "registers," meaning that a chip can internally keep track of three values we can use in a program: A, X, and Y. Registers can also be thought of as variables.

A, or the "accumulator," is used for calculation and most logic. **X** and **Y** are "index" registers typically used for counting and for controlling how we access memory. X and Y can be incremented or decremented but are otherwise less flexible than the accumulator. Each register holds only 8 bits—zeros or ones—of information, corresponding to a number between 0 and 255. (Sometimes a "signed" integer, from -128 to 127.)

6502/6510 "instructions" are all three letters long. To give an idea of what a machine language program is doing most of the time when it runs, here are the most common ones.

- **LDA**: LoaD into A
- **LDX**: LoaD into X
- **LDY**: LoaD into X
- **STA** / **STX** / **STY**: Store content of A/X/Y in a memory location, such as in RAM or to control the graphics and sound chips
- **INX** / **INY**: INcrement X/Y (add 1)
- **INC** / **DEC**: INCrease and DECrease the value of a memory location
- **DEX** / **DEY**: DEcrement X/Y (subtract 1)
- **BNE**: Branch if Not Equal—jump to a location if the last result was not zero, or if a comparison was not equal. This is typically used with CMP or DEX/DEY to create loops.
- **CMP**: CoMPare A to a value
- **ADC**: ADd to A with Carry
- **TXA** / **TYA** / **TAX**: Transfer from X to A, from Y to A, etc. Often you will perform calculations with A and transfer the results to X or Y to adjust the length of a loop.

How can we make a scrolling game? In BASIC, smooth scrolling is impossible, as a BASIC program will take several seconds to move the characters on screen. In assembly/machine language, the same is

straightforward. To create a simple horizontally scrolling game (figure 2.10), we need to synchronize our program with the raster beam (as discussed in the introduction), move the screen one pixel at a time using the scroll register, and move the characters on the screen every eighth pixel. To make the game, we can use the VIC-II's convenient facility for detecting whether a sprite has collided with characters or other sprites. The game can be made in a few pages of programming, and with machine language, the C64 lends itself easily to all kinds of games based on moving objects (sprites), scrolling or non-scrolling backgrounds, and sound.

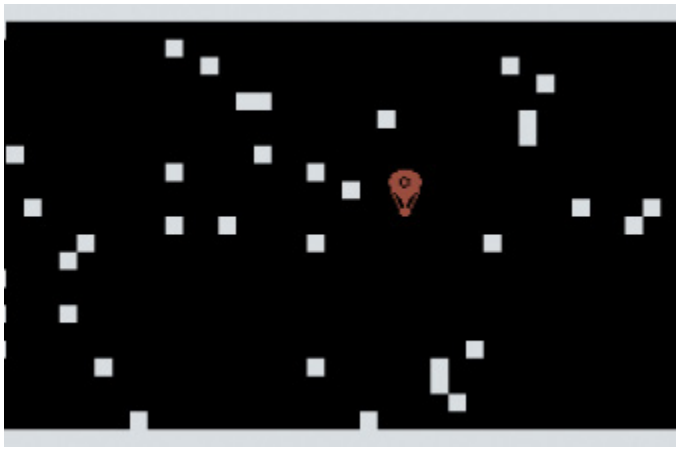


Figure 2.10 Simple scrolling game written in machine language.

► See BASIC and machine language programs for scrolling games on the website.

The Width of an Instruction

6510 programming is verbose, but—like later RISC CPUs—it is correspondingly flexible because each instruction does so little. The 6510 CPU runs at 1 MHz, meaning that it will perform one "cycle" of processing 1 *million* times per second. An experienced programmer will remember the exact number of cycles each instruction takes—from two to seven.

The whole C64 is controlled by the clock of the VIC-II chip, and a 6510 cycle takes the exact time that the TV's raster beam takes to move one character to the right, so to a programmer, the very abstract notion of

1 millionth of a second becomes an entirely bodily experience of how narrow or wide an instruction is on the display.

On the demoscene (chapter 3), counting cycles became central to programming graphical tricks.

How the C64 Escaped from the Arcade Game Tradition

I can cover only a fraction of the at least six thousand C64 games published, but there is a pattern to observe. The C64 was designed in part for arcade games and brought these games to the home, yet in doing so helped all but destroy the economic model of arcade cabinets. Then what? Table 2.1 lists the core conventions of arcade games and the way C64 games rejected these conventions over time. I have organized the games in this chapter around popular games that used and then modified the conventions of the arcade game, and I analyze how they worked with (or around) the design of the C64. There are games—and the practice of "cheats"—that move away from the three-lives limit, games that use arcade conventions with new and surreal content (*Revenge of the Mutant Camels*), games tied to other cultural forms (*Frankie Goes to Hollywood*), and games that defy categorization, often by being more peaceful, expressive, or not based on controlling a character (*Little Computer People* and *SimCity*). C64 history also has adventure games (*The Hobbit*, *The Hitchhiker's Guide to the Galaxy*, and *Maniac Mansion*). Perhaps most uniquely, the combination of arcade conventions and adventure exploration found its most popular form in action-adventure platform games (*Monty on the Run*) and scrolling games (*Boulder Dash* and *Paradroid*) in the first half of the 1980s. Finally, towards the end of the commercial life of the C64, developers created both detailed simulation games and action games mimicking newer computers and home consoles. During this time, each new kind of game also created a new idea of what kind of machine the C64 was.

Table 2.1: How Commodore 64 games built on, and moved beyond, arcade game conventions.

Arcade game conventions	**C64 games moving beyond 1981 arcade conventions**	**Examples**
Simplistic 2D graphics	Improved animation 3D	*Summer Games* *Elite*
Heroic role	New and surreal content	*Revenge of the Mutant Camels* *Hover Bovver* *Manic Miner*
Small universes	Action-adventure games for exploration	*Monty on the Run* *Jet Set Willy*
	Scrolling games	*Boulder Dash* *Paradroid*
	Open worlds	*Elite*
Three lives	Cheats giving infinite lives	All games!
	Other life systems	*Impossible Mission* *Paradroid*
	Games not based on a single goal	*SimCity* *Psychedelia*
Physical/violent conflict	Games about caring	*Little Computer People*

Arcade game conventions	C64 games moving beyond 1981 arcade conventions	Examples
Game-specific universes	Cultural references	*Frankie Goes to Hollywood*
Action and time pressure	Adventure games with (usually) infinite time	*The Hobbit* *The Hitchhiker's Guide to the Galaxy* *Maniac Mansion*

A Machine for High-Quality Graphics and Sound

In the famous early game, an overly dramatic voice greets you, "Stay a while, stay forever!" *Impossible Mission* (1984; figure 2.11), conspicuously easy to confuse with the *Mission: Impossible* film franchise, is a US platform game best known for its initial voice sample. It exemplifies a C64 game with high production values and a Hollywood-inspired setting. The *Computers & Video Games* review emphasized the audiovisual quality of the game, "The sound effects are outstanding. Realistic running and jumping noises are produced throughout the program and *Impossible Mission* also contains some of the most astonishing voice synthesis ever produced on a home computer. . . . The graphics and animation are of the same high quality."[35]

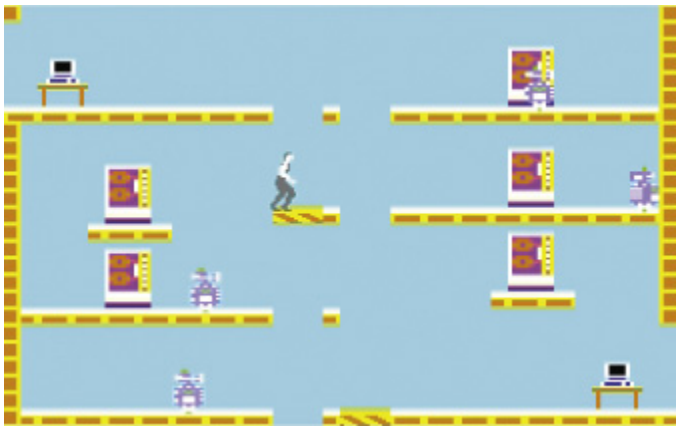


Figure 2.11 *Impossible Mission* (Epyx, 1984), an early platform game with randomized levels, famous for its animation and the “Stay a while, stay forever!” voice sample.

The voice sample was shocking because the C64 was not designed with any feature for playing sound samples. As we will recall, the C64 only has facilities for playing tones according to a frequency. However, the *Impossible Mission* developers exploited a glitch in the C64: changing the overall volume of the sound chip produces an audible click, a louder click the larger the volume change. In BASIC on early C64s, these two commands will produce a louder and a quieter click.

```
POKE 54296,15
POKE 54296,8
```

This technique produces clicks of 16 levels—producing enough clicks quickly allows the machine to play samples with a 4-bit resolution. Disassembling *Impossible Mission* reveals extensive manipulation of the volume register. Also note that the glitch was mostly fixed in later C64 revisions, making this method of playing samples impractical.

The animation of the main character stood out for its fluidity. Designer Dennis Caswell has described how the animation of the main character was the starting point for the game, created from a library book about athletics.[36] The main character broke out of the 24-by-21 pixels bounds of the C64 sprite system and is composed of three sprites—dark legs, dark hair and torso, and arms. The game also moved away from the three-life arcade

model, and instead punishes failure with the loss of ten minutes of the six hours the game grants the player before a deadline expires.

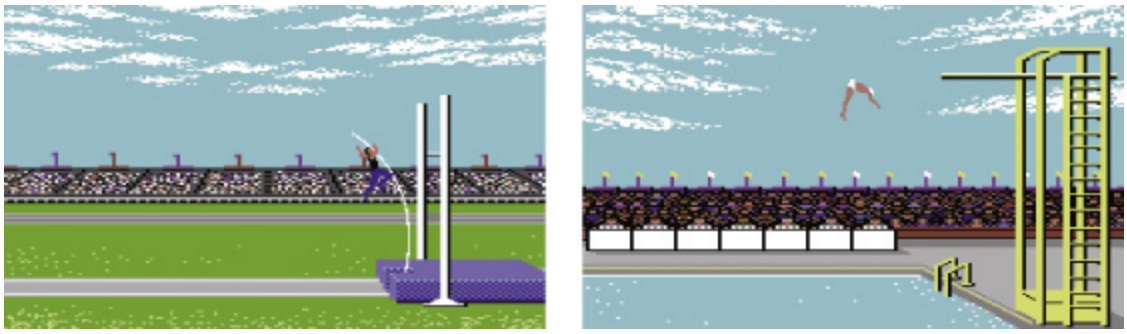


Figure 2.12 *Summer Games* (Epyx, 1984) was impressive for its wide range of disciplines.

The game is modern in its randomization of the levels (the game was explicitly inspired by *Rogue*). Like other games of the time, it is excruciatingly challenging, but this was not considered to be an issue as the challenge just made the game longer—you had to retry the game over and over—giving perceived value to the buyer. Caswell admits that the game was too hard, "I didn't fully appreciate how much more difficult it was for players who didn't know the workings of the game as intimately as I did."[37] This demonstrates a fundamental difference between contemporary video game development and development in 1984: today it is inconceivable for any but the most experimental game to forego playtesting, but for C64 games, this was not a focus.

Summer Games (1984, figure 2.12) and its successors *Summer Games II* (1985), *Winter Games* (1985), and *California Games* (1987) were defining games of the early C64 period, all developed by Epyx Games. Several things stand out in the original *Summer Games*. Though the game is completely different from *Impossible Mission*, it has similarities in being from the same publisher, in the multiple sprites used for the main character, in the heroic roles that players take on, and in the high production values that impressed reviewers and buyers at the time of its release.

As seen in figure 2.13, the C64's sprite system is used to interesting effect, such that the runner lighting the Olympic torch is comprised of four hi-res sprites on top of each other, which makes for more detail than if multicolor sprites had been used. Furthermore, the four sprites of the runner

and the four sprites of the flame are topped with eight flying doves, for sixteen sprites in all, an early example of how to show more than eight sprites on the screen, known as *sprite multiplexing*, discussed in chapter 3.[38]

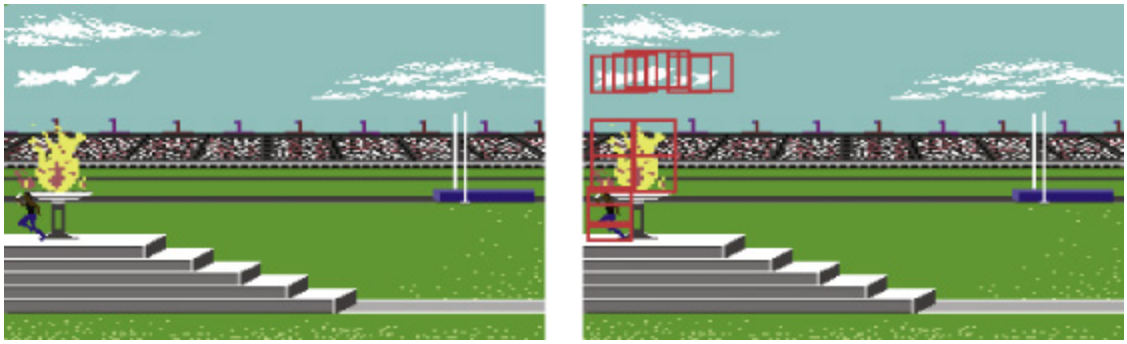


Figure 2.13 Left: The *Summer Games* (Epyx, 1984) opening ceremony was remarkable for the detail and inclusion of a sequence not integral to the game's action. Right: Illustration of the sprite layout.

It is also worth thinking about the playing situation *Summer Games* was designed for. There had been earlier Olympics-style video games like the 1983 arcade game *Track & Field* / *Hyper Olympics*, but *Summer Games* took this home and used the new playing situation to make a much longer and more drawn-out experience for one to four players, competing in eight different events, including the pole vault, platform dive, 100m dash, 100m freestyle, and skeet shooting competitions. Compare this to *Track & Field* with its fast-paced shifts between disciplines. *Summer Games* therefore took part in the game industry-wide realization that the business model of the arcade no longer applied, and that it was possible to make games that included action and moving graphics without putting time pressure on players or quickly forcing a GAME OVER.

An interesting aside—*Summer Games* was developed in the US for NTSC computers, and the platform diving discipline did not work properly on European PAL machines. According to legend, the PAL version of *Summer Games* that Epyx published in Europe was adapted from the "cracked" version—one with the copy protection removed—by German cracker JEDI, and thus the illegal work of a cracker became part of the commercial release.[39]

► See game videos on the website.

A Machine for Surreal Games You Wouldn’t Find in Arcades

The impulse towards copying existing arcade games faded gradually, though some retained the outer structure and conventions of the arcade game but with entirely new content. Consider the works of Jeff Minter.

Figure 2.14 UK developer Jeff Minter’s *Attack of the Mutant Camels* (Llamasoft, 1983), inspired by the *Defender* arcade game and the Atari 2600 *Star Wars: The Empire Strikes Back*. Arcade game conventions with surreal content.

UK developer Jeff Minter demonstrated a love for arcade game tropes combined with surreal or mundane settings. His games also demonstrated how developers only gradually came to understand the C64’s abilities. His 1983 *Attack of the Mutant Camels* (figure 2.14), clearly modeled on the Atari 2600 game *Star Wars: The Empire Strikes Back* (the first *Star Wars* video game, in turn modeled on *Defender*), takes the heroic *Star Wars* setting and remakes it with the titular mutant camels.[40] However, the game scrolls the background only one character at a time, creating a jittery experience. The 1984 follow-up *Revenge of the Mutant Camels* game implemented the intended smooth-scrolling facilities of the C64. The gameplay follows arcade structure, but the content is what game writer Tristan Donovan calls *British surrealism*.[41]

Especially in the UK, C64 games were offbeat and zany compared to both arcade games and more polished titles from the US and (later) Japan.

This appears to reflect both the culture of British humor with its tendency to poke fun at authority and the industry's situation: the initial small-scale development, the lack of industry consolidation, the absence of development for game consoles establishing "quality control" that would erase edgy humor, and a general feeling of being an underdog compared to the US and Japanese game industries. The C64 thus became a machine for creating irreverent, surreal, and psychedelic video games, full of puns, in-jokes, and reversals of video game expectations.

In Minter's 1983 *Hover Bovver* (figure 2.15), players are tasked with mowing their lawn and do this by borrowing the lawnmower of their neighbors one by one, representing a life system. The game features arcade game tropes like an energy meter, synth sounds, and flashing colors, but the actions the player performs are no longer the regular arcade actions of fight or flight, but of cutting grass and avoiding flower beds, neighbors, and dogs. Minter's games thus exhibit a distinct combination of arcade conventions with surreal (or surreally mundane) settings. (► Try *Hover Bovver* on the website.)

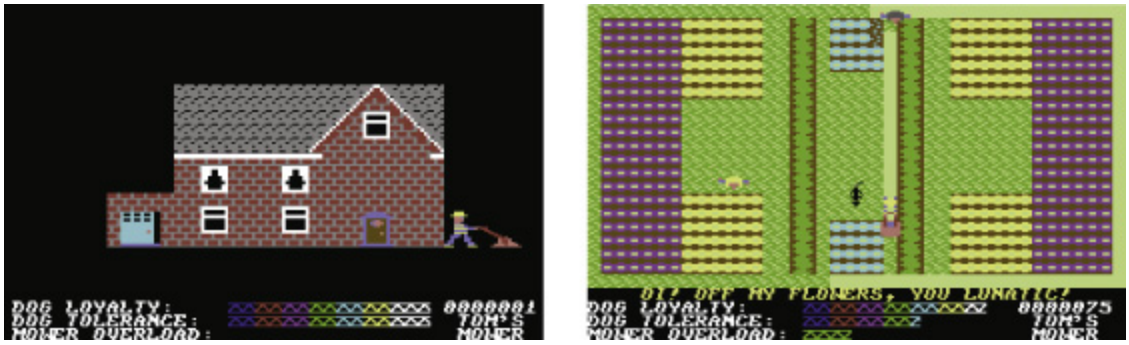


Figure 2.15 Jeff Minter's *Hover Bovver* (Llamasoft, 1983), a British surreal game combining arcade game tropes with the mundane task of moving a lawn.

In this imaginary, the C64 is a device for surreal, psychedelic, and irreverent video games not available in the arcade and not from US or Japanese developers.

A Machine for Adventure Games

Arcade games may have heavily inspired C64 game design, but the other large influence was the adventure game. Adventure games all the way back to 1976's *Colossal Cave Adventure* represented an entirely different set of conventions. They rarely have time pressure, and early adventure games were text-only, running on mainframe computers. They were first freely shared and later sold in boxes, and thus never had a three-lives model and never depended on action or time pressure like arcade games did. Adventure games are also different from arcade games in that adventure games can be completed, whereas arcade games are traditionally designed to become ever more difficult. Though arcade and adventure games are as different as can be, their merger was central to the history of C64 games. But let us first look at three C64 adventure games (figure 2.16), each with its own history.

The Hobbit (1983, 1982 on the ZX Spectrum) was an early adventure game based on J. R. R. Tolkien's book. First developed by Australian Melbourne House for the ZX Spectrum, the game has a surprisingly sophisticated understanding of input, including the ability to give commands to nonplayer characters in the game. Programmer Veronika Megler has described how she disliked adventure games for their static nature, where each game character had a preset location and puzzles were solved with predefined commands. In response, she designed a much more flexible game system.[42] *The Hobbit* presents a dynamic world where characters move around and the game can be played in any number of ways (some of them entirely illogical but amusingly so, such as getting a score higher than 100 percent).[43] I would argue that *The Hobbit* is also graphically more advanced than other graphical adventures of 1982, in part because color was well supported on the new ZX Spectrum and C64 machines.

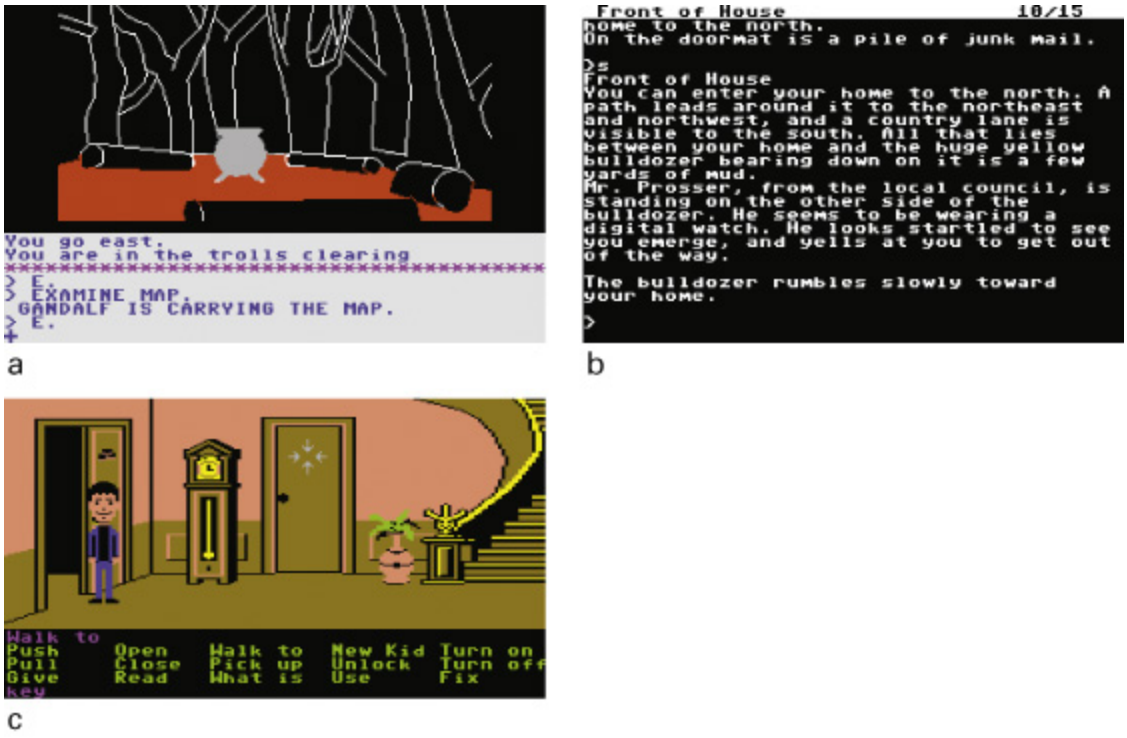


Figure 2.16 Adventure games on the C64: a) *The Hobbit* (Melbourne House, 1983), a surprisingly flexible and dynamic world; b) *The Hitchhiker's Guide to the Galaxy* (Infocom, 1984), based on the book, which was already a parody of adventure games; c) *Maniac Mansion* (Lucasfilm Games, 1987), a multicharacter game where typing is replaced by selecting words.

In *The Hitchhiker's Guide to the Galaxy* (1984), codesigned by author Douglas Adams and game designer Steve Meretzky and based on the book of the same name, the game's form is experimental in a way that mirrors the source material. *The Hitchhiker's Guide to the Galaxy* game was published by Infocom, which also created earlier text-based games like *Zork I* (1980). Infocom could publish its games on a plethora of platforms for two reasons. First, its games were text-based in the teletype terminal tradition discussed in the previous chapter, such that its games only required a computer with text ability. Second, Infocom developed games for its own "Z-machine" format, and designers therefore only had to design the game once but could immediately publish it on many different machines.[44] In 1984, the *Hitchhiker's* game was already commenting on and parodying the text adventure genre, sometimes lying to the player, sometimes playing with impossible objects like "no tea," and the whimsical nature of the source

novel thus made it into a whimsical adventure game. The game even featured a footnote system.[45] While Infocom spent years proudly declaring that text was superior to graphics, the text adventure genre ceased to be commercially viable during the 1980s and was replaced by graphical adventures like *The Hobbit* and *Maniac Mansion*.

Figure 2.17 *Manic Miner* (Software Projects, 1983), the beginning of the UK platform game tradition, launched at a time of political unrest and later coal miner strikes. Having traversed the game's twenty levels (with three lives), it starts over like arcade games do, yet *Manic Miner* was discussed as if it could be completed.

Maniac Mansion (1987) was originally developed for the C64. The playable characters in the game are C64 sprites (each consisting of several sprites). Not originally recognized as a major game for the C64 despite positive reviews, it was only years later that it came to be considered a classic. It is notable for its multicharacter gameplay and the quality of its writing. It was followed by *Zak McKracken* (1988). Like Infocom, developer Lucasfilm Games had an internal format, SCUMM, for porting games to many platforms.

A Machine for Action-Adventure Platform Games

There is a separate UK and European platform game tradition that begins with Matthew Smith's highly successful 1983 *Manic Miner* (figure 2.17), a brutally difficult game for the ZX Spectrum, converted to the C64 and other platforms. As game researcher Alex Wade has observed, Britain was a

divided nation in the early 1980s, moving from "an industrial society based around production, to a postindustrial society based on services and consumption," and this struggle, including strikes and job losses, provided a background for the focus on coal miners in many games of the time.[46] *Manic Miner* consists of twenty levels (then referred to as "screens"), each of which requires the collection of objects to unlock an exit before air runs out. *Manic Miner* was officially inspired by the 1981 *Miner 2049'er*.[47] *Manic Miner*'s structure is subtly different from Nintendo's arcade game *Donkey Kong* (1981), which is (mostly) about getting to an exit to save a woman, rather than about collecting things.

To play *Manic Miner* is to wonder what is really going on, as each of the twenty levels contains an assortment of surreal objects—flying telephones, moving toilets whose lids go up and down, living flowers. The official story is that a group of automatons are living in an abandoned mine, but I am not sure it matters—the game is just surreal and fun. The game's fiction is what I call an *incoherent world*—it is entirely possible to construct backstories about the game events, such as that automatons are flying around a room, but more likely we will just think that our task is to avoid bumping our character into other flat objects on the screen.[48] As a game originally developed on the Spectrum, the C64 version uses the same monochrome objects, but where the moving objects were updated in bitmap on the Spectrum and subject to flickering, they have been remade as sprites on the C64.

Manic Miner set a template for platform games, but that template changed as player habits reformed in response to the new situation of playing in the home. Players noticed the change before developers did. *Manic Miner* followed the arcade game template by providing only three lives and restarting after the completion of the final level 20, in the vein of *Donkey Kong* before it. Nevertheless, users played and discussed it as if it could be completed once and for all. *Manic Miner* won a 1983 Golden Joystick Award for Best Arcade Style Game, but a review from the same year discussed that, "One of the qualities which make a game a winner is whether you'll keep coming back for more. Manic Miner scores well here as it will *take some time to complete*" (my emphasis), and a 1989 a review of the budget rerelease stated that "Winning means negotiating all twenty caverns and getting out at the other end."[49] Though it was designed as an

arcade game with indefinite playtime, *Manic Miner* thus came to be seen as an adventure-type game that could be completed. The follow-up game, 1984's *Jet Set Willy*, could then actually be completed. *Jet Set Willy* retains the surreal character of *Manic Miner* but allows the player to traverse a world of connected rooms, backtracking, and exploring. In the absurd storyline, Willy must clean his house after a party before going to bed in the master bedroom. Should the player achieve this, Willy runs to the bathroom instead and—throws up.

Figure 2.18 *Wanted: Monty Mole* for the C64 and ZX Spectrum (Gremlin, 1984), developed in parallel. The C64 version (left) scrolls and the ZX Spectrum version (right) flips between screens.

To give an idea of the speed of the game industry at the time, *Jet Set Willy* was converted from ZX Spectrum to C64 in four weeks by Shahid Ahmad, who had been promised £3,000 for the conversion, £4,000 if he met the deadline, which he did.[50]

The *Monty Mole* series, starting with the 1984 *Wanted: Monty Mole* (figure 2.18) is a case study in both the specific UK and European game tradition, the relationship between platforms, and the gradual move away from arcade games. The C64 and ZX Spectrum versions were developed in parallel and are completely different games.[51] The C64 version is a scrolling game with multicolor graphics as afforded by the platform; the ZX Spectrum version has monochrome graphics objects and switches between screens. Like *Jet Set Willy*, both games change the platform genre to be a game about nonlinear world exploration, rather than a linear set of screens. This was not lost on the developers, as the C64 version's title screen

explicitly instructed players to “remember its [*sic*] not just a platform game its [*sic*] an adventure.”[52]

Wanted: Monty Mole launched in 1984 during the British coal miners’ strike and featured—in the ZX Spectrum version—Arthur Scargill, head of the striking miners as antagonist. In this first game, Monty must defy the striking workers and steal coal for his family, seemingly anti-strike and anti-union, but developer Peter Harrap has explained that he was for the strike, as his father was a miner, but against the way Arthur Scargill had started the strike without a vote.[53] Confusingly, the publisher allegedly offered 5 pence to the miners per copy sold, and publisher Ian Stewart later described it as being “PR stunt” rather than a political statement.[54]

Figure 2.19 *Monty on the Run* (Gremlin, 1985) was graphically similar to the ZX Spectrum version but featured some of the best-known C64 music.

The idea of combining action and adventure games is often ascribed to the 1980 *Adventure* for the Atari 2600. The combination of platform games and exploration was much later named Metroidvania, from the 1986 *Metroid* and *Castlevania* games for the Nintendo Entertainment System, but these came out several years after the C64 games I am discussing here.[55] Home computer and console traditions were almost entirely separate in the mid-1980s, and Europe’s early video game tradition was partially separate from that of the US and Japan. *Impossible Mission* designer Dennis Caswell claims to have never played *Manic Miner*, which would have been just about impossible for a European C64 developer of the time.[56]

Developers and reviewers took care to explain the new ideas of scrolling and action-adventure. The 1984 *Son of Blagger*, also an early scrolling

platform game, featured an inlay map in the box, marking the "typical image area viewed at any one time" with a rectangle to explain scrolling to unfamiliar players. A review patiently explained this new concept of scrolling to readers, contrasting it with games like *Manic Miner*: "What makes it special is that the room through which your character is wandering is much bigger than the screen itself. The screen is simply a window on to the room which scrolls in four directions to keep the character always in the centre of the picture."[57]

1985 *Monty on the Run* (figure 2.19) is the best remembered game in the *Monty Mole* series with Monty navigating a world of connected screens, like in *Manic Miner*, a strange world of mundane objects—the glass moving back and forth, the smiling monster moving up and down. In *Monty on the Run*, Monty is escaping from the authorities (the story is confusing). Incidentally, *Monty on the Run* launched at the same time as Nintendo's *Super Mario Bros*. did in the US and Japan. This latter game is often presented as the beginning and lodestar for platform games with its exploration and varied levels, but in Europe, the NES *Super Mario Bros*. game was not launched until 1987, giving European platform games plenty of time to develop on their own.[58] In the European tradition, the 1985 *Monty on the Run* was part of an already multiyear history of platform games, of which one could feel jaded, one reviewer reportedly "groaning loudly when [they were] forced to sit down and play what seemed to be yet another boring ol' platform game."[59]

Where the first *Monty Mole* game was entirely different between the two platforms, the *Monty on the Run* versions are graphically similar, the developers having been told that the C64 version "had to be exactly the same."[60] In other words, the *Monty* series was characterized by a continuous back and forth between the ZX Spectrum and the C64, as the UK market at this time was dominated by these two machines. *Monty on the Run* entered the UK software charts at number one in November 1985 (figure 2.7).

► See game videos on the website.

A Machine for Music: The Soul of the C64

> “In the earliest games of 1982 and ’83, most of the music was done by the programmer—the programmer did everything in those days—and some of the music was just absolutely diabolical. It was embarrassing. It was like a drunken monkey on the piano trying to play ‘The Blue Danube.’ I thought, there has to be an opening for somebody who can at least get the notes right and in the correct order, for God’s sake.”[61]
>
> —C64 musician Rob Hubbard

There was one significant difference between the ZX Spectrum and C64 versions of *Monty on the Run*: the C64’s superior sound capabilities enabled much better music, and music was central to C64 culture. From 1986 to 1990, *Zzap!64* featured not only a monthly Readers’ Games Top 30 but a Readers’ Music Top 10, entirely separate from the games in which the music had appeared (figure 2.20). Music was for many users a central attraction of the C64. Pirated collections of music ripped from their games were circulated independently from the games in which they originated, like the 1985 *Close to Me* collection combining music from *Monty on the Run* and *Crazy Comets*. (► Listen to music on the website.)

Figure 2.20 *Zzap!64* published not only a Readers' Games Top 30 but also a Readers' Music Top 10 (1988), showing that music was considered significant content on the platform, irrespective of its origins.

The C64's SID sound chip has similarities to analog synthesizers, and, as a package, the SID chip is close to the late-1970s integrated synthesizer Roland SH-7. Comparing the SID chip to the later Nintendo Entertainment System (NES), we can see that the designers of the NES chip have made many more decisions for the developer. The NES has five voices, but two are dedicated to pulse/rectangle waves, the third is a triangle sound, the fourth is a noise channel, and the fifth is for sampling. The mere three SID voices can be individually set to any waveform and change to new waveforms at any time during playback. This is known as *oscillator sync*. The NES chip may be better for the specific instrumentation anticipated by the designers, but the SID allows for a broader range of expression.[62] The SID was not used for synthesizers like designer Bob Yannes had hoped, but chip designers Yannes and Charpentier later founded the synthesizer company Ensoniq.

Listening to early C64 music like the background music of *Hover Bovver* or the national anthems of *Summer Games*, there is a feeling of stiffness. In the early years, it was common for programmers to create the music for games. Designer/programmer Tony Crowther describes his process as "doing sheet music, I just typed it in."[63] This gave early C64 music a mechanical delivery, playing the notes without any attempt at interpretation in strength or timing.

Compare this to the *Monty on the Run* music, a dramatic piece that shifts between different "instruments," seemingly an accordion, a violin, and a guitar. The *Monty on the Run* music was composed by Rob Hubbard, one of the best-known musicians for the platform. The C64 was home to a handful of other famous composers, like Ben Daglish, Jeroen Tel, David Whittaker, Martin Galway, Johannes Bjerregaard, and Chris Hülsbeck. C64 musicians generally worked independently of game productions and delivered a piece of music on spec. Programming decisions were also compositional choices, and though the C64 hardware was unchanging, the SID sound chip continued to be used in new ways, meaning that music tools continued to evolve and that famous C64 composers were both good programmers and good musicians.

The *Monty on the Run* music demonstrates this clearly: the developers originally wanted Charles Williams's pop-classical piece "Devil's Gallop," famous in the UK as the theme music for the *Dick Barton—Special Agent* radio serial, but Rob Hubbard instead composed a new piece of music heavily inspired by the earlier piece.[64] Furthermore, a new programming technique (pitch bending) became inspiration for the guitar-like middle piece, "[*Monty on the Run*] was supposed to be this run around like a madman type music and has lots of energy. The middle section was an excuse to use the new pitch bend code that I wrote for this project."[65]

The *Monty on the Run* theme is both a composition and programming. Watching an oscilloscope representation of the *Monty on the Run* music, with each line illustrating one hardware voice, we can see how voices often switch between different waveforms, and how pulse waves are manipulated on the fly and change shape, allowing a voice to change timbre while maintaining the same pitch (known as *synth sweep*). (► Listen to *Monty on the Run* music on the website.)

Hubbard's music player separates the program interacting with the SID chip from a table with the musical notes. In Robert McSweeney's 1993 analysis of the *Monty on the Run* music and programming, he notes that Hubbard uses his music routines with only slight modifications across a range of C64 games.[66] McSweeney explains how the music data is based on overall building blocks: three "tracks" corresponding to the three voices of the SID chip, each track composed of "patterns" that can be reused, patterns consisting of notes that can bend, continue a previous note or change instruments. Instruments in turn determine the waveform, vibrato and how a sound fades in or out (the "ADSR envelope").

The data structure of Hubbard music thus corresponds to general ideas in sequencers and music creation. A BBC documentary has Rob Hubbard showing a file of what seems to be the source code for the *Commando* music:[67]

```
!byte $81, $03, $32, $81, $00, $39, $03
!byte $39, $03, $39, $03, $07, $39, $05
```

Although the SID chip was conceptually similar to synthesizers of the time, Hubbard's music practice was very different as the music had to be typed.[68] Hubbard's music notation is textual and based on individual notes rather than chords that involve more than one note at a time. Because Hubbard's format does not adhere to standard note lengths or single waveforms, it is flexible for making expressive music.[69] For example, McSweeney identifies Hubbard's typical bass drum as playing a noise wave for 1/50th of a second, followed by a square wave that quickly drops in pitch.

In 1999, Hubbard explained how the limited number of voices helped to create his characteristic rhythmic style, in combination with inspiration from 1980s popular music like funk and the English electronic music band Depeche Mode. Hubbard said his style was "dictated by the fact that you have such limited resources . . . it's easier to . . . make it sound a lot fuller and like you're doing a lot more if you use much shorter, more rhythmic sounds."[70] Karen Collins has noted how much early C64 music was a reinterpretation of existing music, as is the case with *Monty on the Run*.[71]

The identifiable character of C64 music stems from its limitations—the three voices, cut down from the planned four voices during production, is,

on paper, too few for Western music, and forces a hyperactive style of each hardware voice playing the role of several musical voices. In this way, the limitations of the machine, unplanned, pushed musicians to develop a specific C64 style, as did the common practice of updating the music once per frame of the TV signal.[72] *Had the SID chip had the planned four voices, C64 music would have been less characteristic*.

Chris Hülsbeck's 1986 Soundmonitor (figure 2.21) was among the most popular C64 music editors. Rob Hubbard's music was made without tools, but Soundmonitor is a bona fide editor. Its logic is similar to Hubbard's music routines, but it is interesting because it was published in German *64'er Magazin* as five full pages of hexadecimal data in small font for readers to type in.[73] The magazine's editors clearly believed that Soundmonitor fulfilled a desire for the magazine readers. Soundmonitor prefigured later common tracker software for the Amiga and other platforms. Compared to Hubbard's format, Soundmonitor is closer to sheet music, as it encourages regular note lengths whereas Hubbard's tool allowed nonstandard and custom note lengths. Soundmonitor is a good fit for older Western music but less amenable to jazz (or any kind of music that swings).[74] (► Try Soundmonitor on the website.)

Figure 2.21 Soundmonitor (1986). From musician Chris Hülsbeck, a music editing program for a wider audience, no programming required.

A Machine for Scrolling Games

A central facility of the C64 hardware is the ability to smoothly scroll one pixel at a time, and this set the C64 (along with the Atari 8-bit line) apart

from other contemporary home computers. Scrolling gave a distinct flavor to many C64 games, supporting games with larger objects, larger worlds, and more exploration. I already noted scrolling games like *Wanted: Monty Mole* and *Revenge of the Mutant Camels*.

The 1983 *International Soccer* (figure 2.22) was one of the more successful games developed by Commodore itself. One of the early games to use smooth scrolling properly, it was also new in having a 3D (isometric) perspective, and the relative intelligence of the players stood out. Like with *Summer Games*, the inclusion of unnecessary details like the players entering the field made the game feel elaborate and extravagant. The use of scrolling also mimics television broadcasts of soccer, which only show part of the field. (► See scrolling game videos on the website.)

Figure 2.22 *International Soccer* (Commodore, 1983), an early successful game from Commodore, impressive for unnecessary details like players entering the field.

Boulder Dash (1984, figure 0.3), originally developed for Atari 8-bit computers, came about as a tortured collaboration between developers Peter Liepa and Chris Gray.[75] *Boulder Dash* was an early C64 favorite, in which you must collect a set number of diamonds per level before making it to the exit. The game did not stand out visually, but physics engines were uncommon in 1984, and the addition of gravity and falling objects to a puzzle environment felt new. One review described the rocks as "positively Newtonian."[76] *Boulder Dash* is also a scrolling game. The inclusion of

scrolling happened for a roundabout reason: the original character was too small, as Peter Liepa explained:

> When I showed an early version of the game to a potential publisher, they pointed out [that] “the man” was way too small, and [needed] to be a more recognizable character. But I couldn’t make “the man” more prominent without making everything larger as well. So this is where the hard work began of converting the game from one that ran on a 24x40 character display to one that scrolled over a much larger region. The caves still had 40 elements across, but each element was now made up of 4 characters arranged in 2x2 pattern. So each cave was now 80 characters across, and scrolling was introduced. . . . I was much more interested in animating him, and I think that the way he blinks his eyes and taps his feet when you are not controlling him added a lot of depth to the character.[77]

Scrolling thus serves not just to create large playing areas but to enable larger and more expressive game elements.

Figure 2.23 *Choplifter!* (Brøderbund, 1982) for the Apple II. Though the platform lacks scrolling facilities, the combination of small foreground and background objects moving at different speeds is enough to give the appearance of movement and depth.

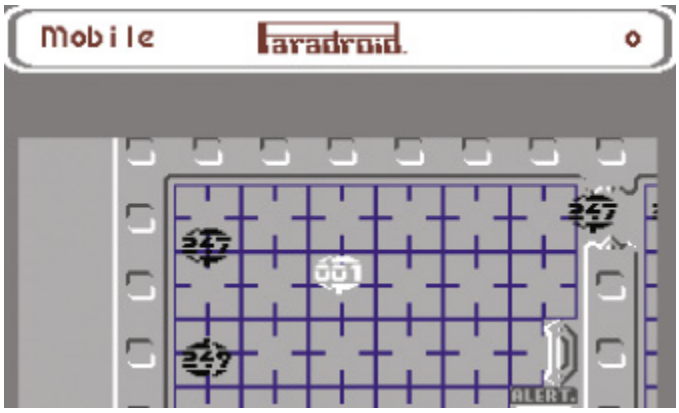


Figure 2.24 *Paradroid* (Hewson, 1985), a classic game of no clear genre, featuring embossed graphics and a combination of action and puzzles.

There is one important note about scrolling: computers without scrolling facilities could still give the impression of moving around a space by setting the game in a world where it made sense for most of the pixels on screen not to change, paradigmatically by using a monochrome background. In 1982's *Choplifter!* for the Apple II (figure 2.23), most of the screen is an unchanging black sky and purple (!) landscape, and the sense of traversing a world is created by having small objects on the lower fifth of the screen move at different speeds to give the feeling of depth and movement, though only a few pixels are being updated.

The 1985 *Paradroid* (figure 2.24) by Andrew Braybrook is found on several lists of "the best C64 games of all time."[78] Like many other early C64 games, it does not belong to any identifiable genre. You control a small "influence device" labeled 001 with the goal of taking over robots. You move your robot around a scrolling world, and when faced with another robot, a "transfer" minigame gives you the opportunity to take it over. The goal of the game is to clear all spaceship floors of robots.

Like some of the other C64 games discussed, *Paradroid* is odd in that it will take hours to complete yet offers no way of saving your current position or state. It does not have a life system, but as the player takes over higher- and higher-level robots, being shot or failing a minigame gets the robot ejected, down to the original 001 robot, where failing will lead to GAME OVER. *Paradroid* was heavily promoted by *Zzap!64* which featured a development diary by Andrew Braybrook. In the first installment, Braybrook was already distancing himself from the new idea of

the arcade adventure: “It’s not really an arcade adventure—it leans more towards arcade.”[79]

Paradroid was new in its visual style. Earlier games had used graphics elements that were generally flat (like the legs of the runner in *Impossible Mission*), but *Paradroid* adopted an embossed style for environment and maps, where the multiple levels of grey in the C64 palette are used to give the impression of a light source shining from the upper-left corner, signaling that objects stand out from the background or are recessed into it.

Scrolling games were among the most characteristic C64 games. Scrolling games are not a *genre*, given that the player does vastly different things in each one, but they do share a feeling of exploration, of the world being larger than what we can see. Other important C64 scrolling games include *Neoclyps* (1983), *Sheep in Space* (1984), *Cauldron* (1985), *Commando* (1985), *Sanxion* (1986), *Iridis Alpha* (1986), *Katakis* (1986), *Delta* (1987), *Wizball* (1987), *R-Type* (1988), and *Turrican* (1990). Some developers specialized in scrolling games, like UK Tony Crowther, whose games were characterized by horizontal scrolling used for different types of games, like the previously mentioned *Son of Blagger* (1984) and *Wanted: Monty Mole* (1984), in addition to *Killer Watt* (1984), *Gryphon* (1984), *Loco* (1984), *Black Thunder* (1985), and *Suicide Express* (1985). Rival developer Jeff Minter even criticized Crowther for relying too much on scrolling.[80]

A Machine for Cheating

Monty on the Run retained the three lives of the traditional arcade game and was—like other platform games of the time—unbelievably difficult to complete. Today, the assumption is that a game of exploration should offer unlimited lives or retries as well as the option of saving your progress. But in 1985 there was no convenient, universal way of saving a game. You could create a save system, but this only became popular on disk drives some years later, and password systems for accessing levels were also a late innovation. As such, playing these games today juxtaposes interesting game exploration with the ill-fated structure of limited lives.

Figure 2.25 *ZZap!64*'s June 1985 POKEs for *Manic Miner*, allowing users to modify the game as they see fit.

As ideas of saving one's progress and of gameplay in general developed, players often used cheats when playing games—either through POKE commands published in magazines, like the POKEs for *Manic Miner* shown in figure 2.25, or cheats added to a cracked version of the game. Cheats were a way to modify a game to match changing ideas of how video games should work. In her study on cheating, Mia Consalvo identified "getting stuck" as players' primary reason for cheating.[81] POKEs were published regularly in *Zzap!64*, *64'er Magazin*, and *Aktueller Software Markt*, and in

the *Gold Mine* column of US *Commodore Magazine*, but they were generally more common in European publications.

Where do POKEs come from, and how would we find one ourselves? Recall that the POKE command changes one byte in the machine's memory, meaning that we can use POKEs to change any program in the C64's RAM. If we want to be immune to collisions with monsters in *Monty on the Run*, we can consult the *Commodore 64 Programmer's Reference Guide* and see that the VIC-II chip has a convenient facility for detecting sprite collisions for the programmer and always shows recent collisions in the $D01E memory location.[82] We can search for references to this location in the program and find the following commands, the only reference to the sprite collision location.

```
$0C0E LDA $D01E
$0C11 STA $48
```

Apparently, the *Monty on the Run* program loads then saves the sprite collision status in the $48 location for later processing. No problem, if we change the second command to LDA $48, such that the value is never stored, we can walk though enemies. We can then tell our friends to make the corresponding change as POKE 3089,165, send the POKE to a magazine, or publish a cracked version of the game with a cheat option. Even if the average user did not disassemble the program, the proliferation of POKEs made it clear to users that computer (game) programs were things you could modify yourself. This C64 imaginary was about being part of a culture of copying and modifying software, modifying the way a game was played.

A Machine for Multimedia and IPs

In the 1980s, it was unclear what relationship video games had to other kinds of popular culture. There were tie-in games to TV shows, toys, and especially movies including *Barbie* (1984), *Ghostbusters* (1984), *Top Gun* (1987), *Miami Vice* (1986), *Highlander* (1986), *The Rocky Horror Show* (1985), *Commando* (1985), *Batman* (1989), *Terminator 2* (1991), and *The NeverEnding Story* (1985). Many of these came across as cynical and

would often replicate the action of a movie in a well-defined action genre, or, in the case of *The NeverEnding Story* (1985), recreate the plot as a series of adventure game puzzles.

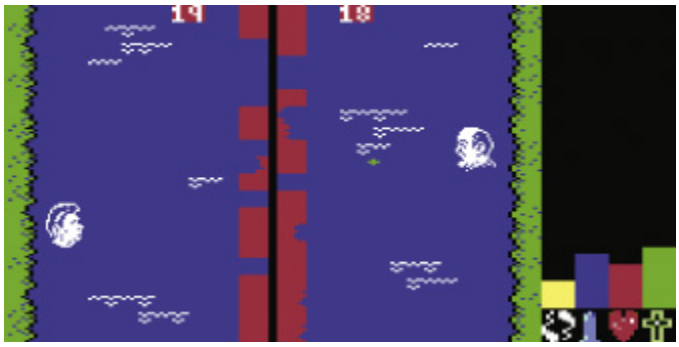


Figure 2.26 *Frankie Goes to Hollywood* (Ocean, 1985) was truly strange and interestingly congenial to the 1980s synth-pop band. This image shows Reagan and Gorbachev attacking each other.

There was also a small tradition of music tie-ins in the 1980s, where it was understood that a C64 computer could replicate a musical experience or the tone of a band. For example, the Paul McCartney-driven movie *Give My Regards to Broad Street* spawned a same-titled game (1985), in which the player drives a car around London to find missing friends and missing tapes accompanied by a maddening looped version of the Wings song "Band on the Run." The 1989 *Michael Jackson's Moonwalker* was less experimental and just combined existing genres.

The 1985 *Frankie Goes to Hollywood* (figure 2.26) game is more interesting in it represents the titular 1980s synth-pop band and has both a freer and more congenial relationship to its source than has *Give My Regards to Broad Street*. As the manual states, "Frankie has sent you over 60 tasks in your journey from Mundanesville through the Pleasure Dome," referring to the band's first album *Welcome to the Pleasuredome*. Your overall task, the often surreal settings, and the mixed genre of the game connected well to the band's image. (► See video on the website.) For Ocean Software, licensing was a deliberate strategy to get press coverage, and to be able to approach computer manufacturers for bundling their games with computers.[83]

Not That Great a Machine for 3D Games

The C64 was designed around the two-dimensional arcade games of 1979–1981, and 3D graphics were not part of its original design goals. However, 3D games began to appear. *Elite* (figure 2.27), one of the best-known 3D C64 games, originally released for the BBC Micro in 1984, was radical in three ways, two technological and one in overall structure. Technologically, it broke ground by showing proper, opaque 3D objects (with hidden lines removed), and by its procedurally generated universe, where each solar system and planet had its own description. Structurally, the game was revolutionary in its open world structure which did not force the player to perform any given task but allowed for free roaming in the large universe. (► See video on the website.)

Figure 2.27 *Elite* (Braben and Bell, 1985) is a 3D space game in a procedurally generated galaxy. The C64 was not designed for 3D graphics, but the flickering spaceships were impressive. Rejecting a core component of arcade game design, *Elite* was new in creating an open world and not forcing the player to play for a specific goal.

The C64 version of *Elite* was inferior to the BBC Micro original. The BBC Micro lacked the C64's dedicated hardware for sprites and scrolling, but this was largely irrelevant in 3D games, and the BBC's faster CPU allowed for a better 3D experience. These computers were decidedly not designed for 3D games, and clever techniques were required to perform the necessary math.[84]

Elite arguably started the "open world" genre of games and created the expectation that the C64 could be used for 3D. The C64 appeared to become ever more capable, so why wouldn't there be many more and even better 3D games for the C64? Later 3D C64 games included *Mercenary* (1985), *Rescue on Fractalus!* (1985), and *Driller* (1986), which had fully colored 3D objects at the expense of the display, which only updated two times per second.

A Machine for Genre-Defying Games

Apart from adventure games, the games so far have had some relation to the action-oriented games that the C64 was originally designed for, even if some games had longer playtimes. As with *Elite*, some developers began to renege on traditional game structure, or gave players optional goals.[85] Though the title was a bit on the nose, Jeff Minter's 1984 *Psychedelia* (figure 2.28) broke entirely with the game format in being a "light synthesizer" with no goal. It appears to have been inspired by John Conway's *Game of Life*, as a grid structure where each cell is influenced by adjacent cells to create an ongoing and unpredictable pattern. This principle was used in later music visualizers, and Minter was not one to downplay the allusion to drug culture: "It was certainly something that I used a lot—many spliffs were smoked and much Pink Floyd listened to whilst using Psychedelia, and at computer shows we'd always play music whilst I would demonstrate the program to passers by."[86] (► Try *Psychedelia* on the website.)

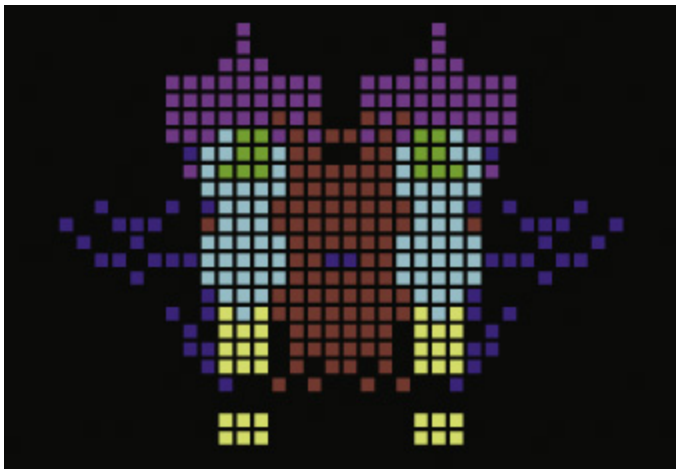


Figure 2.28 *Psychedelia* (1984), Jeff Minter's "light synthesizer" with no game structure. According to Minter, "many spliffs were smoked and much Pink Floyd listened to whilst using *Psychedelia*."

A more game-like title of the time was 1983's *Moondust* by Jaron Lanier (famous for popularizing the term "virtual reality"), which shares with *Psychedelia* a focus on generating new visuals. The 1986 *Tubular Bells* took this further and packaged a C64 remake of Mike Oldfield's 1973 album of the same title, accompanied by generated visuals. These three titles share the fact that they were highly experimental and distributed as games. These developers saw the C64 as a platform for audiovisuals.

Two experimental C64 games played a role in some of the biggest franchises in video game history, *The Sims* and *SimCity*. The 1985 *Little Computer People* (figure 2.29) is based on the conceit that little people live inside your computer. You can interact with the little man and type commands such as "play piano," which are sometimes executed, sometimes ignored. *The Sims* (2000) was officially inspired by *Little Computer People* and works in the same vein of tending to virtual people with their own wills and goals.[87] *Little Computer People* thus created a different relationship between the player and the action on the screen than in the arcade or in the other games discussed so far.

Figure 2.29 *Little Computer People* (Activision, 1985) provided an early experience of interaction (through text) with a character "in your computer" who does not always do what you ask, inspiring *The Sims*.

Figure 2.30 Will Wright's early C64 game *Raid on Bungeling Bay* (1984), designed around the scrolling features of the C64, later inspired the development of *SimCity*.

Alongside *Siren City*, the 1984 *Raid on Bungeling Bay* (figure 2.30) is perhaps the least famous game with the most historical impact. *Raid on Bungeling Bay* was developed by Will Wright, who would later go on to design *SimCity*, *The Sims*, and *Spore*, but *Raid on Bungeling Bay* led concretely to *SimCity*. *Bungeling Bay* is a helicopter game of flying around a map, attacking enemies, bases, and factories. It was the special scrolling facilities of the C64 that led Wright to the *Raid on Bungeling Bay* design. "I was trying to find things on the Commodore you couldn't do on the Apple II," he recalled. "The Commodore 64 had these graphic features that were advanced for what it was and you could have this big scrolling window on a

larger world. I also always loved helicopters. I designed that game around the technology, around what you couldn't do on the Apple II."[88]

According to Chaim Gingold's book *Building SimCity*, Wright was working on a robot game but found himself preferring to edit the *Bungeling Bay* map with his custom tool and later developed this into *SimCity* on the C64 (figure 0.3). Inspired by Jay Forrester's *Urban Dynamics* and cellular automata, Wright used the C64 character mode as the basis for an increasingly complex city simulation.[89] The central unit of *SimCity* is known as a *character* in the source code for other platforms, revealing that the game was originally designed around this C64 hardware feature. Furthermore, *SimCity* has seven disasters (known as "objects" in the manual), corresponding to the seven sprites left once the cursor is accounted for, and the game's logic is structured around a division between regularly recurring interrupt routines (from C64 raster interrupts) and programming executed outside those interrupts.[90] The game's publication was tortured, as publisher Brøderbund found it difficult to understand why anyone would play an open-ended simulation, repeatedly asking Wright, "when would he be making an actual game out of the simulation? You know, something with a winning state, perhaps with a computer opponent to play against?"[91]

Brøderbund dropped the project in 1985, and it lay idle for two years until businessman Jeff Braun took interest and founded publisher Maxis to port the game to the Amiga and Macintosh, hoping their interfaces and mouse control would improve the game. It was eventually released for the Macintosh, Amiga, C64, ZX Spectrum, and MS-DOS in 1989. By that time, though, Wright had lost the source code for the C64 version, and Gingold describes this version as a historical document of an early state in the evolution of *SimCity*.[92] C64 reviewers were puzzled by the omissions of features from the Amiga version, like the budget and evaluations screens that we now know were only added later in the development process.[93] (► See videos on the website.)

The Technical European Games and US Simulation Games

Arcade design remained even as the arcades were losing importance. In the second half of the 1980s, developers had a firm grasp of the C64 hardware and could create games with a level of detail and movement far beyond early games. European developers in particular learned from the demoscene tricks discussed in chapter 3 or were themselves active on the demoscene. For example, *Turrican* developer Manfred Trenz expressed his admiration for the demoscene.[94] One of the lessons from the demoscene was that you did not have to assume—like in *Wizard of Wor*—that the background or landscape should be made with characters, and the moving objects should be made with sprites. You could combine them in any way you wanted. (► See videos on the website.)

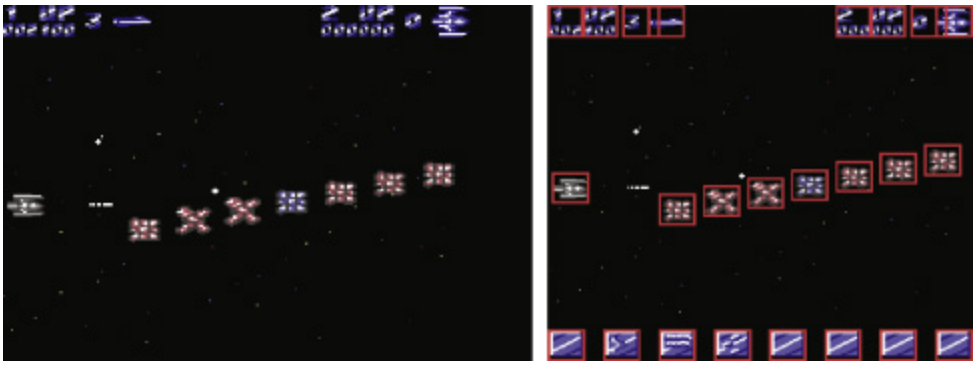


Figure 2.31 *Delta* (Fasoulas, 1987) kept the three-lives model from the arcade but used newer techniques for showing sprites in the top and bottom borders. Almost all moving graphics are sprites; characters are only used for the starfield and for shots.

The 1987 *International Karate+* (figure 0.2), or *IK+*, was probably the peak of martial arts games for the C64. It followed Australian *The Way of the Exploding Fist* (1985) and the original *International Karate* (1986). *IK+* thus wasn't the first martial arts game for the C64, but its level of detail and polish stood out. You can play with three players at the same time (up to two human players and one computer-controlled), achieved with judicious sprite multiplexing and building the blue fighter out of character graphics instead of sprites.[95] Even if the action was inspired by the arcade, this home game was set up to allow much longer play sessions.

Delta (1987), shown in figure 2.31, is a shoot 'em up apparently inspired by Japanese games like *Gradius/Nemesis*, in addition to Finnish developer

Stavros Fasoulas's earlier *Sanxion* (1986). *Delta* features a demoscene technique of opening the top and bottom borders, to be discussed in chapter 3, here used for showing score and upgrade status. Counter to what we might expect, most graphics are sprites; only shots and the background starfield are made from characters. Rob Hubbard's title screen music is closely derived from the Philip Glass score for the 1982 movie *Koyaanisqatsi*.[96] Incidentally, Philip Glass commonly uses arpeggios to create his trademark minimalist classical music, creating a close match to the abilities of the SID sound chip.

Delta is polished, visual, and aurally impressive, but its three-lives per game structure missed the lesson of switching to the home. In practice, likely, many players used cheats, often on pirated copies, to progress in the game.

Figure 2.32 *Last Ninja* (System 3, 1987) is an isometric arcade-adventure game with puzzles and visual depth, where objects can be partially covered by scenery.

1987's *Last Ninja* (figure 2.32) was perhaps the pinnacle of the action-adventure hybrid on the C64, combining action, exploration, and puzzles in a visually attractive package, where the isometric 3D world allows objects to be behind and in front of each other. This is not 3D in a modern sense, but the sense of depth, coming at the great technical cost of modifying the sprites when hidden by objects in the world, gave the game a modern feel. It also featured music by Ben Daglish, one of the most popular pieces of C64 music outside of *Monty on the Run*, and led to follow-ups and conversions to other platforms.

The C64 action-adventure platform game tradition continued and gradually introduced larger and more varied worlds. 1985's *Cauldron* switched between platforming and flying, and 1986's *The Sacred Armour of Antiriad* built on this combination, adding larger characters and high-quality graphics. It is also striking how Nintendo's 1986 game *Metroid* arrived at a formula remarkably similar to simultaneously released C64 games like *Antiriad*. The C64 tradition continued through the 1990 German game *Turrican* by Manfred Trenz (figure 2.33), an action-platform game with shooting and many technical tricks. This game features a large and varied world that can be explored semi-linearly, large "boss" monsters (made with multiplexed sprites), and parallax (two-layer) scrolling made by modifying the character set on the fly, giving the impression of a distant plane moving at slower speed than the foreground. Another example of the C64–Nintendo tradition merge, the level layout of the 1987 *The Great Giana Sisters* borrowed heavily from *Super Mario Bros.*, and distribution was stopped after legal threats from Nintendo.[97] The game has since been remade in more legally sound versions. Trenz was also involved in *The Great Giana Sisters*, and the widely beloved music for both games was made by Chris Hülsbeck.

Figure 2.33 *Turrican* (Rainbow Arts, 1990), one of the last C64 hits, is a German action game with high technical quality and a sense of depth created by animating the background. It also merges the European C64 scrolling action-adventure with the Nintendo action-adventure tradition.

The games discussed so far in this section were all European. I don't mean to suggest that US developers did not have technical skills, they were

just of a different type. Where late European development often focused on smooth and fast-moving graphics inspired by arcade games, US developers were often making technically complex simulation games, mimicking some real-world activity. Accolade developed *Test Drive* (1987) and *Grand Prix Circuit* (1989), among other driving sims. MicroProse, cofounded by famed game designer Sid Meier, developed a range of war simulations such as *F-15 Strike Eagle* (1985) and *Gunship* (1986), but was ultimately most successful with *Sid Meier's Pirates!* (1987) shown in figure 0.3.

Returning to *Maniac Mansion* (1987), it is striking that as a late 1980s game, it missed several important technical C64 innovations. It retains the original C64 font, which other games had forsaken several years prior. It uses the SID chip in a very rudimentary way, just playing the regular waveforms without dynamically modifying the voices. When the screen scrolls, it is jittery and does not use the C64 scrolling facilities. Only the use of sprites is C64-specific. These strange limitations appear to represent the split between European and North American C64 cultures, where North American developers were focusing on other platforms and had not taken in recent lessons about the VIC-II chip. This does not, however, detract from an interesting and funny adventure game.

On the C64, later industry behemoth Electronic Arts published Atari 8-bit conversions like 1983's *Archon* and *M.U.L.E.*, game construction games including *Racing Destruction Set* (1985), role-playing game series *The Bard's Tale* (1985, 1986, 1988), and the adventure game *Neuromancer* (1988). Activision had a wide range, publishing the licensed *Ghostbusters* (1984), hacker simulation *Hacker* (1985), Lucasfilm Games' *Koronis Rift* (1985), life simulator *Alter Ego* (1986), and the licensed *Die Hard* (1990).

US developers of the late C64 era thus focused mostly on slower and more complex games, likely for an older audience. We can speculate that the US game market in the late 1980s had divided into a younger and more action-oriented home console market, and an older audience attuned to slower games on home computers. Interestingly, French publisher and developer Infogrames similarly eschewed fast action in the late 1980s, developing instead graphic novel–inspired games like *Passengers on the Wind* (1987) and *Tintin on the Moon* (1989).

Towards the end of the commercial C64 market, several high-quality arcade conversions meant that C64 charts gained a significant percentage of

Japanese-origin games. The 1990 *Rainbow Islands* was a conversion from the Taito arcade game (1987), a sequel to *Bubble Bobble* (figure 0.3, 1987), and became the game to spend the longest time on the *Zzap!64* readers' charts of the 1990s (figure 2.5).

The changing country origin of games in the UK *Zzap!64* charts (figure 2.6) hints that the C64 lasted longer as a platform in Europe. In general, US C64 magazines shut down three to four years before European ones.[98] *Koronis Rift* designer Noah Falstein explained to me that "the sales of C64 games started dropping so we switched to releasing first on the PC, and as many other studios did the same, that in turn hurt the sales of C64s."[99]

Subsequently, the European market itself also became divided, evident by a comparison of two February 1992 charts, the UK *Zzap!64* top-ten readers' chart and the German *Aktueller Software Markt* (*ASM*) sales top-ten chart.[100] Surprisingly, only one game, German *Turrican II*, appears on both charts. The UK chart contains seven local games including *Creatures* and *Lords of Midnight*. By contrast, the top six spots of the *ASM* chart are German games, including a newer version of *Bundesliga Manager* (based on the German soccer league). The balance between UK and German titles in both markets fluctuated, but it is notable that both the UK and German markets were big enough to be at times dominated by local hits.

A Machine for Piracy

Aarhus, Denmark, 1985

> It is early summer, and we are two friends waiting at my house, a third friend making his way on a moped, carrying an eagerly awaited floppy disk with a game. The game is *Elite* [figure 2.27], and it is obviously pirated. When the game launches to the title and the 3D spaceship begins spinning on my parents' 26″ color TV, it is a new world, it is the future, it is everything we have hoped for. There is a pleasant gift economy of pirated games, it is a special occasion when a friend brings new games to share.

The rumors are true. The history of C64 game culture is intrinsically connected to software piracy. I have so far talked about games as products that came in boxes. In practice though, games were usually received on pirated tapes or floppy disks. According to Gleb Albert, statistics about piracy are contradictory, with figures ranging from 14 percent to 99 percent of software being pirated, based on a many different methodologies, and some of the numbers were part of game industry campaigns against piracy.[101] Although the true percentage of software that was pirated will never be known, it was likely at the higher end of the scale.

How to Crack a Game

How were C64 games copied though? The simplest strategy was to copy the medium itself. Users could copy tapes using regular audio equipment, but the quality quickly degraded to make second-generation copies unusable. Disk copy programs read the bytes on the floppy into memory and wrote it onto another disk. As time progressed, developers created increasingly strange disk formats that regular disk copying programs considered to be errors, so increasingly smart disk copying programs were made, and so on. Laine Nooney has described the history of the Apple II program Locksmith designed to copy copy-protected floppy disks.[102] Such advanced copy programs were plentiful for the C64 but played nearly no role as all software was cracked—the copy protection was removed—within weeks if not days after release, and sometimes even before release due to leaked copies from insiders. According to engineer Bil Herd, Commodore itself was involved in piracy: “Here’s a little known fact: Commodore was probably the biggest source of cracked software. The worst thing you could do was submit a copy of something to the games and applications group.”[103]

When cracking, a program on a non-copyable special-format floppy disk or tape is moved to an easily copied normal-format floppy disk, a crack intro is added (see figure 3.1), and often cheats are made available. Some games used the accompanying material as copy protection: The *Jet Set Willy* cassette inlay featured a 40-by-18 grid of color squares, and the game quizzed users about colors in the grid to allow play (the then-absence of color copiers and digital cameras made it hard to copy such an inlay). Crackers summarily removed the quiz, allowing the game to run anyway.

In a 2013 book about US Gold, the UK publisher of US software, Epyx Games developer Chuck Sommerville claimed that for *California Games*, "the disk version even today has not been pirated due to the excellence of the copy protection."[104] His assertion was met with incredulity and was quickly debunked, as people familiar with C64 cracking knew there was no such thing as unbroken C64 copy protection.[105]

Why not? The basic problem of copy protection is that the bytes on whatever medium you are using must at some point be read into the computer's memory for the program to work. If a cracker can read the bytes and then save them, a copy can be made. In the very beginning, many C64 manuals instructed users to press two keys at the same time, SHIFT—RUN/STOP, which leads to the machine printing "LOAD," and then "PRESS PLAY ON TAPE." Once the program had been loaded, the "RUN" command is given automatically and the program starts. Users soon realized they could in many cases just type "LOAD," and then "SAVE" once the program had loaded, giving perfect copies of a program onto another tape or disk. Even some early cartridges could be inserted, saved, and distributed. The standard quickly became that both tape and disk programs featured "loaders," small programs starting automatically and loading the full program using a faster "turbotape" routine on a custom tape format or loading the program from disk with a "fastloader," often from custom formats and ignoring errors, all this meant to prevent copying. This denied users the opportunity to just SAVE the program to a tape or disk.

How would you crack that, then? The cracker could just get the loader in memory, make it do its job of reading the software from the custom format, and when the game was meant to start, it would instead be saved to a standard floppy disk, allowing future copies to be made. Chuck Sommerville trusted in Epyx's Vorpal copy protection for *California Games*, which relied on custom-modified disk drives to copy and would check if the Vorpal routines were still in memory. His statement suggests a lack of familiarity with cracking practices—crackers could just read the data for the game's disciplines into memory and save it onto a regular disk, removing the checks for the custom loader and disk format. Given a sufficient supply of motivated crackers, the C64 developer was always going to lose. For the end user, the good pirate distribution networks gave piracy a convenience that could hardly be matched by copy-protected store-

bought software. For crackers, games provided an interesting challenge, a way to demonstrate skills, a social environment around cracking, and mostly friendly competition with rival cracking groups. Having cracked games myself at the time, I will not defend the practice as it clearly hurt game developers, but the challenge was tantalizing and allowed me to use my technical skills in a way that non-programmers could appreciate, and to gain pseudonymous credit for my accomplishments, though no monetary gain. In combination, this made life difficult for C64 developers. “I finally gave up on installing copy protection schemes because it was a wasted effort,” Joseph O’Hara from Microtechnic Solutions told me in 2023.[106]

A Machine for the Computing Underground

Like with software development, there were significant differences between countries. Historian Gleb Albert argues that the different starting points of US and European game companies also meant a different relationship to pirate groups, as those who founded and ran the big US game companies like Brøderbund or Sierra were from an older generation than the European “computer kids” who received their first computers in 1982–1983. The US companies had no contact with the cracking scene, whereas West German developers would try to dissuade their cracker friends from cracking their games.[107] The press coverage of Interceptor Software supports this, showing that European developers were often in their teens.[108]

Software piracy was dealt with inconsistently between countries. The European Union did not standardize on copyright protection until 1993, but individual countries began efforts to battle piracy earlier, either through legislation or through acts of individual litigation.[109] In Sweden, the first criminal proceedings against a seller of pirated software took place in 1986, the 19-year-old pirate objecting that he was just selling already cracked software.[110] The increasing focus on software piracy during the mid-1980s created concrete, sometimes parental, pressure to move away from illegal activities. Underground *Illegal!* magazine reported nervously about the police raid on the 1988 Madonna Cracking Group party in Marl, Germany, accusing the party organizers of collaborating with the police.[111] In my interview with demoscene participant Swyx, he reported of widespread rumors about police raids in Denmark in 1987 and of hiding all his equipment at his father’s workplace for fear of a raid.[112]

For regular users, pirated software was plentiful and expected. Commodore Scandinavia sales manager Jesper Christensen told me of the complaint of a customer who had bought a Commodore computer in a supermarket along with ten floppy disks, only to be shocked to find the disks empty. The consumer filed a complaint with the Danish equivalent of the Better Business Bureau.[113] The idea that they would have to pay for software, or that floppy disks would come without games, was alien to them.

Figure 2.34 Megacollection 20 Videogames. The early Italian C64 software market was almost exclusively pirated, and newsstands sold collections of pirated C64 games, often translated to Italian. (Photo by Damiano Gerli.)

In Italy, software copyright was not upheld, and pirated games, sometimes localized, such as *Cammelli* (an Italian version of *Attack of the Mutant Camels*) were readily available at newsstands (figure 2.34).[114]

Damiano Gerli reports that since non-pirated software was hard to find, and since pirated software was sold in regular stores, “Most people had no idea they were actually buying pirated software.”[115] The German history of piracy is probably the best documented. In 1987, German company Data Becker declared that it had placed undercover detectives in computer clubs, which may or may not have been true but certainly served the function of warning against software piracy.[116] In a surprising 1988 incident, the German cracking group Radwar participated in a television debate about piracy with controversial attorney Günther Gravenreuth, who famously wrote letters to (presumably) teenage boys who had posted ads requesting to swap programs, pretending to be a teenage girl offering pirated software.[117] A demand for 1,100 German marks was sent to anyone who responded.[118]

Figure 2.35 Classifieds ads of German *64’er Magazin* in April 1984 promising a range of software—most likely pirated—either by “swapping” (*tausch*) or for a modest fee.

A 1994 article in *64'er Magazin* was critical of Gravenreuth's methods, but beyond the duplicity about his identity, it should be said that the classifieds section of the 1984 magazine (figure 2.35) contained many ads promising program swapping (*tauschen*) or software for a small fee (*Unkosten*).[119] By all accounts, this meant piracy. By 1988, the classifieds section was fitted with a warning against piracy and ads offering pirated software for a fee ceased to be published, but some readers were still looking for possibly legal *Tauschpartners*—swap partners.

It was obviously legitimate to swap public domain software or games that you had tired of, but "swapping" was also a technical term in cracking groups—there were programmers, graphics artists, musicians, and "swappers" in charge of procuring software to be cracked and for distributing cracked software and demos. An article in the magazine *Illegal!*, dedicated to piracy and demos, would advertise for *Neue Tauschpartner*—new swap partners—in 1987 with undisguised illegal intent.[120]

As the name *Illegal!* suggests, there was, in some corners, a fascination with (and self-promotion of) being underground. *Illegal!* presented cracking as cool, connected, and daring. The 1988 *Illegal!* issue #30 reflected on how the cracking scene started, expressing nostalgia for the first cracked games the reader saw on the C64, "THINK BACK! What was the very first game you ever saw on your 64? Probably 'COMMODORE SOCCER' which was cracked by a German guy called 1103, who also cracked other Commodore modules. Or 'M.U.L.E.' from Electronic Arts cracked by 'OLEANDER'? Or 'HARD HAT MACK' by OTD? Or 'SUMMER GAMES' by JEDI?"[121]

Why crack games and why make demos? Patryk Wasiak describes cracking and later demo-making as the "computer underground" and shows that it worked as one *youth lifestyle* of the 1980s, analogous to graffiti or clubbing. The computer underground also shared with graffiti and underground clubbing a fascination with illegality and handles. Wasiak sees cracking as a way of "building adolescent masculinity"—in conjunction with media panics about hacking and cracking, illegal computer skills became highly desired and worked as initiation into a culture. As Wasiak notes, "in public discourse, the figure of a hacker is seen as a lonely geek who lacks social skills and interactions, but being a cracker required

advanced social skills to establish and hold a position in a peer group."[122] This is parallel to the observations I make in chapter 3 about the social nature of the demoscene.

In the beginning of this chapter, I referred to Graeme Kirkpatrick's argument that the game culture of the 1980s gradually became separated from computer culture, eschewing programming terms and developing its own language for assessing games.[123] However, the prevalence of piracy on the C64 worked in the other direction, presenting programming skills as attractive and as the central skill for providing a continuous supply of free games. Chapter 3 shows how the demoscene also worked, sometimes in conflict with game culture, to give status to programming skills.

Impossible Missions and Absent Damsels

The C64 graphics chip was targeted at the techniques of popular arcade games of 1979–1981, making it easy to create fast-moving arcade games, and to later move beyond the conventions of those arcade games, creating slower games, explorative games, adventure games, and games entirely outside arcade genres. As new video game genres appeared, the C64's biggest limitations were the poor facilities for 3D graphics and the lack of a good universal facility for saving game progress. The C64's scrolling facilities were tied to its own tradition of action-adventure games.

Some developers were slow to accommodate the new playing situation in the home, but POKEs and the game modifications that cracked games often contained worked as unofficial workarounds for harsh game design practices, like the three-life structure of the arcade when it stuck around in large exploratory games where it made little sense. Only at the end of the 1980s did expectations shift. Of the 1989 *Indiana Jones and the Last Crusade*, the *Compute!* reviewer wrote, "Since you can't save the game, you must work your way through each level every time you play."[124]

Seen from today's ideas about game design, most early C64 games come across as positively *broken*. We would chastise game design students for turning them in as projects, and the steep difficulty curve or impossibility of completion would be caught in playtesting. Game design has come a long way since. *And yet*, players reported—and still report—deep satisfaction with these C64 games. Who are we to tell them otherwise? Today we expect

that a video game meant to be completed should be completable by most players, at least by spending enough time. But in the early C64 years, it was acceptable for a game to be impossible. If players wished, there would be POKE, a cheat, an adventure game walkthrough, or a cracked version to fix the problem, making these seemingly broken game design practices perfectly workable.

In terms of the fiction (or story) of video games, it has been observed that video games, including 1980s video games, are often built around the "damsel in distress" trope where a male hero rescues a passive female.[125] Noting that the existing work on the topic focuses mainly on console games in North America, Maria Garda and Paweł Grabarczyk compared top retro charts of the NES and C64. It turned out that on these charts, 33 percent of NES games contained the damsel in distress trope, but—surprisingly—only 5 percent of the most popular C64 games did.[126] Garda and Grabarczyk propose several tentative explanations, including the possibility that Nintendo was deliberately targeting younger audiences to whom the trope would have greater appeal, or that the less tightly controlled nature of C64 software development meant that C64 games were made by a larger and possibly more diverse group of developers. Additionally, I would argue that because most C64 games were either too difficult to complete or not meant to be completed, they were structurally ill-suited for traditional narratives like the damsel in distress. Finally, the irreverent tone of UK games in particular meant that C64 games rarely featured the conventional heroes you would expect to be saving a damsel.

Influence and the Transatlantic Computer

It is too simplistic to measure culture and art by its influence on later culture and art, but the C64 was the platform for which most games were made in the 1980s, and it is interesting to consider its effects on game history. C64 games were experimental, wild, and could borrow freely from the arcade and the adventure game. UK games in particular were surreal and political, and there was little need to streamline production. In my previous book, *Handmade Pixels*, I wrote about how the idea of "indie" or "alternative" games only appeared in the late 1990s.[127] 1980s developers reported that, even though they were making strange experiments on small

budgets, they were *just making video games*, in any way they could. Many early independent games of the 2000s referred specifically to Nintendo games, and independent games often contain a nostalgia for a simpler time, where one person or a group of people could create the video game they wanted, rather than being beholden to a big corporation. However, rather than Nintendo, it was the home computer—and especially the C64—that was the actual precursor to present-day, small-scale, experimental game development.

In terms of design, there are four game types and three game franchises for which the C64 played a leading role:

1. The **open-world game**. Games where players are free to explore a larger world and make their own choices. Though *Elite* was originally a BBC Micro game, it was popular on the C64, and *Sid Meier's Pirates!* similarly allowed players to traverse a world without being pressured to work towards a goal.
2. **The death of game over**. Though developers took a long time to adjust to the realities of home gaming, the C64 influence is also a practice of adjustable difficulty and infinite lives then made available through POKEs and cracks but now integrated into game design conventions.
3. The **action-adventure**. From early platform games like *Wanted: Monty Mole*, the combination of exploration and action was consistently popular on the C64, also enabled by the ability to scroll the world.
4. **Graphical online worlds**. 1985's *Habitat* is often referred to as the first graphical massively multiplayer online game. It was created for the Quantum Link online service to which players would connect with modems.[128]
5. ***SimCity***. The game was originally developed on the C64 and was feasible only due to the C64's character graphics, scrolling, and sprites.
6. ***Sims***-like games. *Little Computer People* was an early game to allow the interaction with virtual characters with their own will, inspiring *The Sims*.

7. ***Grand Theft Auto* (*GTA*)**. Originally a 1997 2D game, *GTA* was inspired by the otherwise bland *Siren City* (figure 0.3), as both involve freely traversing a city of criminals and police. Steve Hammond, one of the original developers of *Grand Theft Auto* (figure 2.36), personally recalls "playing a Commodore 64 game called *Siren City*, a top-down car game with cops."[129]

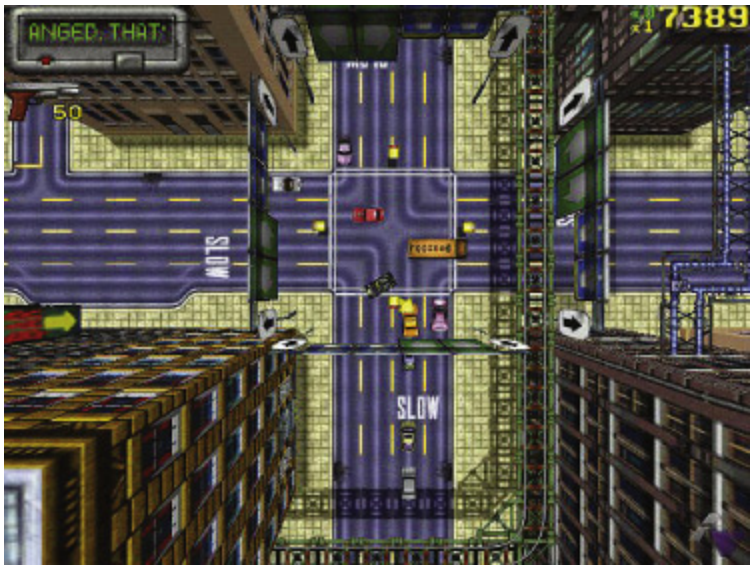


Figure 2.36 *Grand Theft Auto* (DMA Design, 1997). The first *Grand Theft Auto* game was eerily similar to *Siren City*, both top-down games of police chases in a freely traversable city.

Writing this book, I have been surprised to realize just how often 1980s home computer, C64, and European games are omitted from video game and computer histories. It may be due to two compounding oversights: 1) early European game history was largely home computer–based, and is thus neglected in US and Japanese histories, which focus primarily on consoles, and 2) the hacker history of computing focuses on Microsoft, IBM, and Apple, such that even global histories of computing neglect much of computer history, including the C64. It was also surprising to realize just how different the console and home computer landscapes of the 1980s were between the US and Europe. As I have shown, US home computer games of the mid-1980s were made for the Apple II, C64, Atari 8-bit, and IBM, whereas European games were made for the C64, ZX Spectrum, and Amstrad (Schneider) CPC. The C64 was the only popular transatlantic

game computer of the time, but it is striking just how different US and European games were—heroic US games, experimental UK games, and later more action-oriented and technical mainland European games.

3

Against Intentions

Imaginary #3:	The Commodore 64 can do more than we thought.
	The Commodore 64 is a ticket to an underground community.
Programming Language:	Machine language
Graphics:	Custom character sets, multiplexed sprites, border sprites, early 3D
Colors:	Colored objects on black backgrounds, later a wider variety of colors
Quotes:	It was a portion of Bob Stevenson's Tutankhamen picture—which filled the whole screen! —Zzap!64, *December 1986*
	[The party features] an electrician to fix the power supply, so that there will be no fuses blown. *—Invitation, Dominators/Upfront/Trilogy Party, 1989*

	It dawned on me that beyond just being fun, having ambition, and wanting something, we were driven by what others thought of us. This was emphasized by charts for everything—best demo groups, best cracking groups, best everything. *—Interview with demoscene participant Swyx, 2021*
	- Comic (ex 2000 AD) joined ILLUSION . . . - The disc-mag FATAL NEWS from CENSOR DESIGNS will only be published on paper now. . . . - Two norwegian guys called Pi and Control have joined BEYOND FORCE. - Rythem joined WWE. *—Demoscene gossip in* Pirates *8, 1990*
Examples:	jesperjuul.net/c64/3/

The $64,000 Glitch

The C64 comes with no reset button; you just turn it off and on many times a day and face the startup screen with its dark blue background, light blue border, and text. This is perfectly impressive, and you know the border can never contain graphics.

Depending on where you are in the world, the flickering TV shows this image fifty or sixty times per second, always controlled by the Commodore 64 computer. Inside the computer, one of roughly thirty chips (figure 0.10) sends the required signals to the TV, and in that chip, microscopic components keep track of when to show the central text area and when to show the border (figure 0.13). You understand that the canvas is the central dark blue field of 320 pixels horizontally and 200 pixels vertically, and you are proud that your C64 bests both its VIC-20 predecessor with its measly 176-by-184-pixel resolution, the Apple II (280-by-192 pixels), and the ZX Spectrum (256-by-192 pixels).

In 1985 and 1986, it turns out that there are top/bottom and sideborder tricks, as you can fool the chip components into forgetting to show the border—the top layer of the C64 display (figure 0.13)—and instead let sprites and the background color show through (figure 3.1). It is not a bug in the graphics chip, given that you are using the chip in a way not sanctioned by the documentation, it is a *glitch* that feels monumentally valuable. Your machine can do even more than advertised or expected; its pixel resolution is higher than you thought! If you program, your metaphorical canvas has literally expanded, and you have a way to show your skills.

This created a new imaginary: the C64 became a device for cool, interesting, and surprising programming. The border tricks were developed by humans, of course, and they had a social function, but this was also a profound experience that the C64 had a will of its own. I discussed Jane Bennett's argument that things can act by themselves and have an efficacy "in excess of human meanings."[1] The C64 was designed by humans with intentions, though these were quite contradictory, yet it also did things that neither its designers nor its users had anticipated. These new abilities cast the C64 in a new light and the machine pushed C64 culture in a new direction.

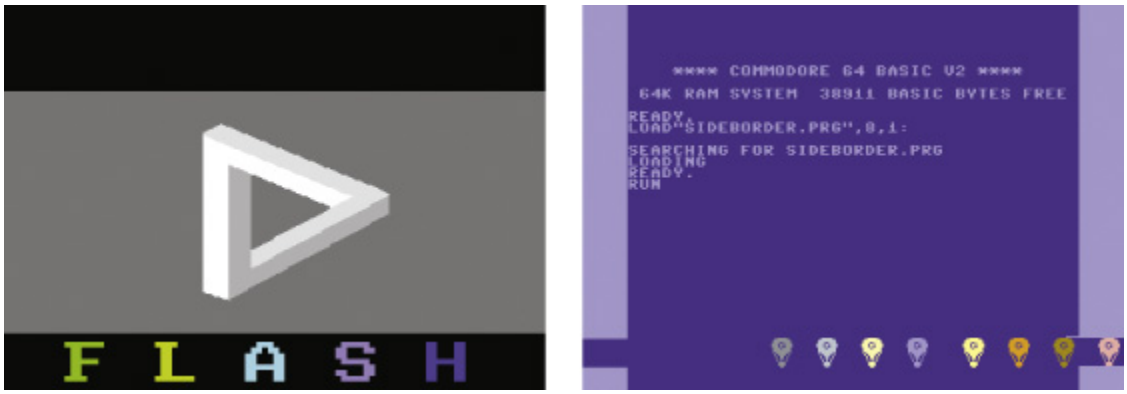


Figure 3.1 *Flash XI* (Flash Cracking Group, 1985) demo. Left: Sprites in the bottom of the screen where only the monochrome border was expected. Right: *Sideborder program* showing sprites in the border.

Figure 3.2 Border sprites! Left: The *ESCOS I* (1001 Crew, 1986) demo disabled the border entirely to fill the screen with text explaining the technique and claiming a trademark on "sideborder sprites." Right: *TUT* (1986) demo showing Tutankhamen in the border. This new technique was a shock to C64 users and central to the demoscene.

In the December 1986 issue of *Zzap!64* magazine, writer Julian Rignall explained his reaction to sideborder demos received from the Dutch 1001 Crew (figure 3.2):

> An interesting parcel arrived at ZZAP! towers the other day . . . Inside was a letter and disk from some "Dutch 64 freaks" called the 1001 Crew. . . . So, we loaded in the first demo, which turned out to be something of a shock . . . It was a portion of Bob Stevenson's Tutankhamen picture—which filled the whole screen! Somehow, the

> 1001 Crew have managed to dispose of the border—and as you hopefully can see from these screen shots the effect is quite stunning.[2]

► See demos on the website.

It is notable that the 1001 Crew presented themselves as a *group*, as if they were a band, with the word *crew* "inspired by the break dance group Rock Steady Crew."[3] Group members were identified by pseudonyms or *handles*. The group was so self-conscious as to make an utterly unenforceable trademark claim, "the words 'sideborder sprites' and 'no-border sprites' are trademarks by the 1001 crew 1986." The group was also sufficiently interested in publicity as to send their demos to *Zzap!64*, and the game magazine—which had disavowed all things technical in its inaugural issue—wrote effusively about the demos, explaining not how the trick was done (that came later) but how fascinating it was. The tricks also presented a problem for a magazine. When printing pictures of C64 software, it was customary to crop away the border, making it difficult to visualize a border effect, and *Zzap!64* resorted to a full photo of a TV running the demo.

Demos were made by the *demoscene*, widely understood to be a technical culture where participants create demos showing off programming skills and new tricks for exploiting the hardware of relevant computers.[4] By most accounts, the early demoscene was most active in Germany and northern Europe in the mid- to late 1980s, centered on the C64, with most participants being young men.[5] I take an open view and include in this chapter anything called a *demo*, even if the label was only applied later.

Rather than write a new history of the demoscene, I will tell the story of one 1989 demoparty and use this to show how the C64 demoscene came to be and gradually changed. I am interested in the striking amount of effort spent on social interaction and organization in the demoscene, and I investigate the fascinating history of C64 technical demo tricks. The demoscene is in many ways strange. Demos are not serious software like word processors, but neither are they entertainment like games, nor does the demoscene associate itself with fine art,[6] so what are demos? This uncertainty leads to some recurring questions about the demoscene that I will explore in this chapter:

The demoscene and industry: Especially in the history of the Finnish demoscene, as in Lassi Tasajärvi's *Demoscene: The Art of Real-Time*, the demoscene is frequently described as feeding directly into gainful employment in tech and the Finnish game industry, and tech companies are often reported as scouting for talent at demoparties.[7] It is true that game development studios such as Remedy (*Max Payne*), IO Interactive (*Hitman*), and Starbreeze Studios (*Syndicate*) grew out of demo groups, but is it characteristic of the demoscene?[8]

The demoscene and technology: Markku Reunanen's PhD dissertation, *Times of Change in the Demoscene*, shows the ostensibly high-tech culture's conflicted relationship to technological change, sometimes sticking to older platforms like the C64.[9] Is the demoscene about new technology?

The demoscene and games: Reunanen also argues that the demoscene gradually distanced itself from games during the 1990s: "The demoscene had become increasingly self-conscious and independent, and drew the line between itself and other hobbyists."[10] Similarly, did game culture distance itself from demos and programming?

The aesthetics of demos: Daniel Botz's German-language *Kunst, Code und Maschine* sees demos as an aesthetic phenomenon and the demoscene as a culture where participants share ideas about what is good or important in a demo.[11] What did the C64 demoscene consider to be a good demo?

The demoscene and piracy: Patryk Wasiak has examined the demoscene's ties to illegal cracking of software, and Gleb Albert has examined the trading networks of pirated software and their relations to the demoscene, including around 1990 when the demoscene expanded into formerly Eastern Bloc countries.[12] How did software pirates and the demoscene interact?

Hacking? Finally, if we look at the preponderance of technical tricks on the C64 demoscene, isn't the demoscene hacking?[13] The *New Hacker's Dictionary* defines a hacker as "a person who enjoys exploring the details of programmable systems and how to stretch their capabilities, as opposed to most users, who prefer to learn only the minimum necessary."[14] I am wary of the hacker angle because, following Steven

> Levy's 1984 book *Hackers*, hacking is often understood as another linear timeline where all computer-related culture in the world is traced back to a single source, adding a few select Californian engineers and erasing the vast majority of computer history.[15] Is it more likely that "hacking"—the creative and innovative use of computers—is continuously invented?

Origins of the Demoscene

In the common telling, cracking groups (as discussed in the previous chapter) created intros for their cracked software (or *cracktros*, figure 3.1) to claim credit, and around 1985 these intros became standalone programs called *demos*.[16] The demoscene retained a connection to software piracy until the 1990s at least, and demos were initially spread using the networks for trading pirated software.[17] I interviewed former game cracker Chix from the Triangle group, who gave up cracking software because his parents did not approve of such illegal activities. Demo-making, on the other hand, was met with parental approval. At least he wasn't cracking software.[18] Trap from the Bonzai group reported that his interest in the demoscene came from a music collection received alongside pirated games: "On one of the tapes, there was a music demo by The Mercenary Cracker. It was the first time I saw something that wasn't a game that I thought was cool, and it was made by a completely ordinary person like me. Unlike the game developers who were untouchable gods."[19] Living in Denmark with nearly no game industry at the time, Trap found demo-making to be the more attainable activity.

In a 2004 interview, developer Honey from the 1001 Crew, originators of the *ESCOS* and *TUT* demos, took credit for the idea of the standalone demo, made to make sure that the new border programming technique was widely noticed:

> **What was the coolest thing someone invented on the C64?**
>
> Naturally, our side-border sprites that were done in early 1986 (Border Letter 1). To make sure the whole world would see it, we decided to put it out as a demo instead of using it in a game intro. (It was new to

bring out a small program purely to show technique, but the word "demo" at this time did not exist.)[20]

The latter claim is not quite true as there is a much longer history of demos designed to demonstrate the features of a platform.[21] Figure 3.3 shows the 1982 Commodore *Christmas Demo*, distributed to Commodore dealers and demonstrating the music and colorful graphics of the machine. (► See demo on the website.) Patryk Wasiak has pointed out that while the *Christmas Demo* was not considered part of the demoscene, demoscene demos were often used by stores to show off the machine.[22]

Most demos, and nearly all C64 demos, emphasize the real-time generation of images as opposed to playback of predefined animations.[23] This is an *aesthetic*, in the sense that most demos are valued for what the programming achieves in real time. As Botz says, "The computer is a not a tool, but the material," but we might as well say that *computation* is a central material of demos.[24]

Figure 3.3 Commodore's 1982 *Christmas Demo*, a demonstration of the C64's abilities used by dealers. Graphics are simple PETSCII characters. Not considered a demoscene demo, but important in C64 history.

The early C64 demoscene was in many ways ephemeral, materials stored on floppy disks, magazines distributed as photocopies, but it is also striking as a culture that thoroughly documented itself. There are systematic databases of the demoscene at large, such as Pouët.net and Demozoo.org. The C64-specific Commodore Scene Database (CSDb) aims to capture every single piece of software distributed for the machine, including all

demos, in addition to other documents from the demoscene.[25] Another archive even contains the mailed letters between demoscene participants.[26] With archival material there is a danger that we take the map for the territory and let whatever is in the archive determine our research, forgetting that we are looking at a mere selection.[27] Instead, we should understand that these archives exists due to an interest in memorializing the demoscene, sometimes personally motivated by a desire to cement one's status in the demoscene, and in the early 2020s, this interest has even led to the demoscene being nominated and/or recognized as a UNESCO intangible cultural heritage in Finland, Germany, Poland, the Netherlands, and Switzerland.[28]

It is useful to start with a historical overview of platforms. Using data from the Pouët archive, I have mapped the popularity of different platforms in the demoscene (figure 3.4), showing a C64 peak in 1988 to be replaced by three following waves, the Amiga (peak 1991), MS-DOS (1995), and Windows (2002).[29] Furthermore, while the demoscene is smaller than it used to be, the C64 has experienced a gradual resurgence since 2010.

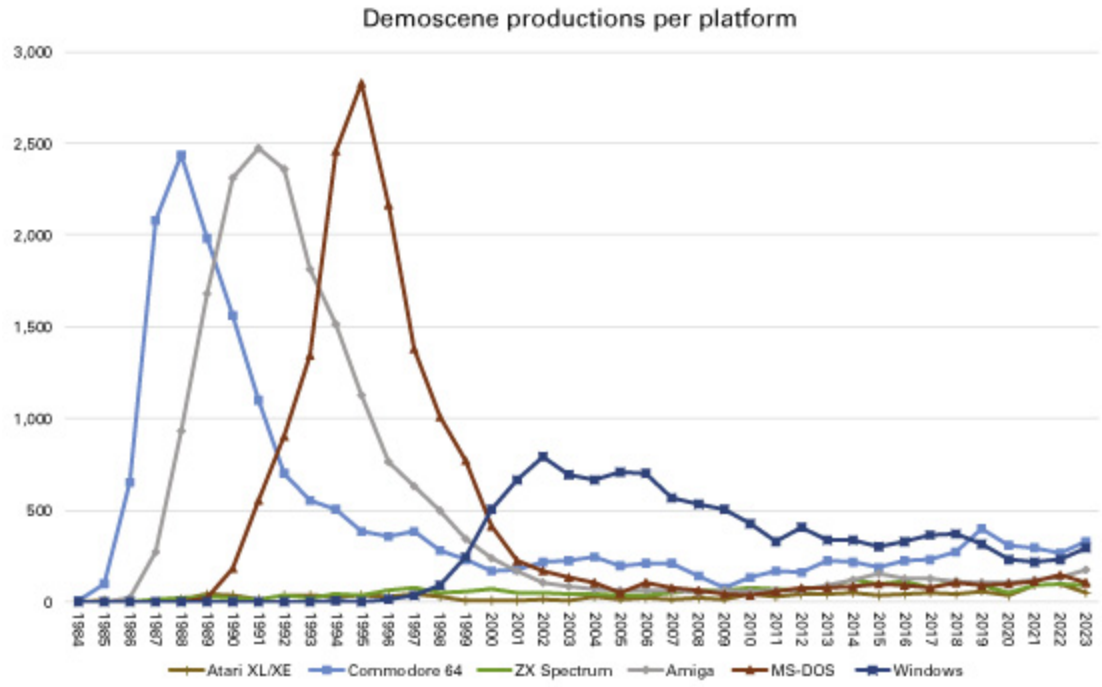


Figure 3.4 Demos released across platforms, 1984–2023, showing that the early demoscene was centered on the C64 as well as the recent resurgence of Commodore 64 demos.

Conventional wisdom says the early demoscene was concentrated in Germany and northern Europe, but is that true? Using the CSDb, I analyzed the geographical distribution of cracks and demos for the C64 (all-time numbers), and I also created figures adjusted for the population of these countries in 1990 (figure 3.5). It is important to note that any number of factors may skew such a database, especially concerning which releases were entered and how demos were attributed to countries. The database indicates that (West) Germany was the overall dominant country on the C64 demoscene and cracking scene, while, adjusted for population, Denmark was the country with the most cracks and demos. At the end of this chapter, I propose an explanation for the country distribution.

Demoparties

Demos were often released at demoparties. But what happened at early demoparties? The CSDb lists the first parties as occurring around 1985–1986. First called a "copyparty," these events were typically attended by between ten and twenty participants, though over time parties slowly grew in size.[30] Photos from the 1988 Jewels/Danish Gold/Dominators/Upfront Party (figure 3.6) show a modest event at a village community center. Attendee Trap reported that "it started in a scout hut. There was only space for 30–40 people, and when I arrived, there were people standing outside with their computers in hand. Eventually, in the middle of the night, they found this community hall, and we all went there in buses. That was when I first met legends like Jeff Smart."[31] This 1988 party became notorious for its disorganization and for the power going, overloaded by too many C64s. Demoparties are now often considered related to the later LAN party—often commercial, large-scale events with thousands of mostly game-playing participants—but the relation between demoparties and LAN parties is tenuous, as there are always more game-players than programmers.[32]

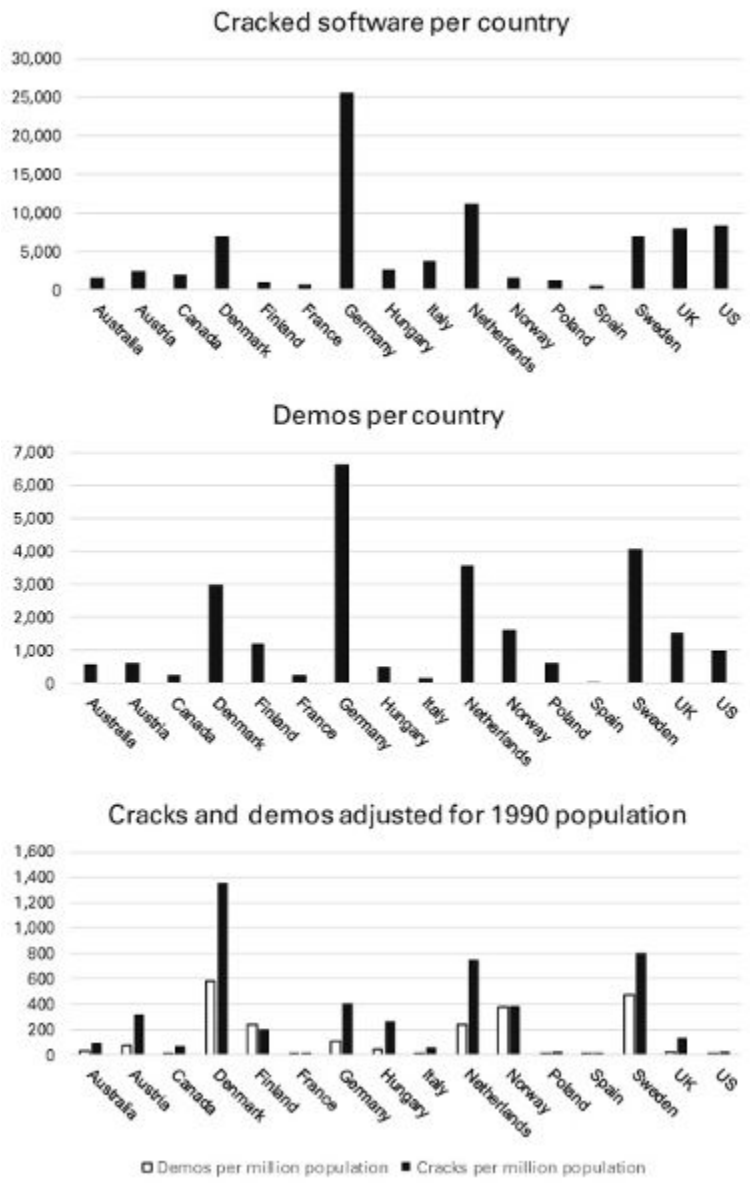


Figure 3.5 C64 cracks and demos in selected countries. For cracks (pirated software), most appear in Germany, the Netherlands, the US, and UK, but for demos, it is Germany, Sweden, the Netherlands, and Denmark; the US is nearly absent. Adjusted for population, Scandinavian countries and the Netherlands are the most active originators of cracks and demos.

Figure 3.6 Top: 1988 Jewels/Danish Gold/Dominators/Upfront Party at a village community center. Bottom: Unpacking C64 equipment from an orange Volvo. (Photos by Björn Fogelberg.)

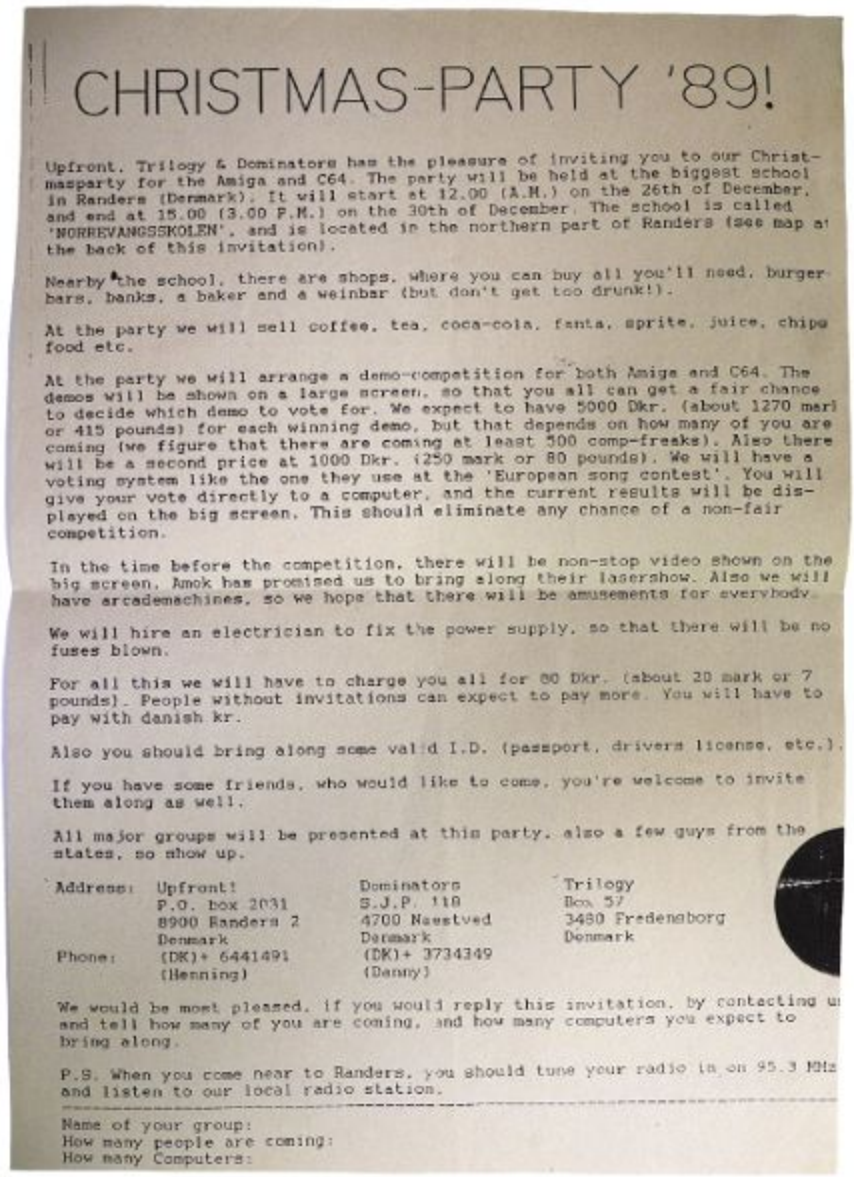
I have studied one specific early party, the December 1989 Dominators/Upfront/Trilogy Party in Randers, Denmark.[33] I chose this because it is bookended in two ways: 1) It was promoted as more professional than previous parties and marked a transition away from earlier private, informal gatherings and/or piracy-oriented “copyparties” in the

demoscene; and 2) At the other end, it had no game-related activities or competitions and was thus also distinct from later commercial LAN parties.

Though demos are, everyone agrees, at the center of the demoscene, the demoscene does not just consist of a few central objects, such as floppy disks with demos on them.[34] A demoparty is machines, electricity, heating, food, transportation, people, and software. I have examined the extant material surrounding the 1989 party, notably demos, invitations, and reports. To complement this, I have interviewed participants from the party, one organizer of the party, and three visitors. As it happens, I also visited the party and entered a demo into the competition. I also look at selected demos from the party to see the claims they were making and the function they served at the party.

The 1989 Randers Party

In 1989, private users did not yet have access to the internet, and though services like UK Compunet and “bulletin board systems” (BBS) had appeared, allowing users to connect their C64s to community message and file-sharing boards, they were not widespread in Europe. As Kevin Driscoll notes in his study of modem culture, using modems to access BBSs or online services was generally expensive in Europe as flat-rate local calls were not common like they were in the US.[35] There were illicit strategies for exploiting loopholes in the phone system, such as calling free numbers to reach a dial tone or using calling cards of dubious origin, but software was generally distributed by physical mail. The preferred method was to cover stamps with scotch tape or lamination, such that the postmark could easily be erased and stamps could be reused for free.[36] Invitations to the Dominators/Upfront/Trilogy Party were therefore spread physically as mailed paper letters or as floppy disks. Figure 3.7 shows the paper invitation for the event (there were also two demos that were invitations to the party).



CHRISTMAS-PARTY '89!

Upfront, Trilogy & Dominators has the pleasure of inviting you to our Christmasparty for the Amiga and C64. The party will be held at the biggest school in Randers (Denmark). It will start at 12.00 (A.M.) on the 26th of December, and end at 15.00 (3.00 P.M.) on the 30th of December. The school is called 'NORREVANGSSKOLEN', and is located in the northern part of Randers (see map at the back of this invitation).

Nearby the school, there are shops, where you can buy all you'll need, burgerbars, banks, a baker and a weinbar (but don't get too drunk!).

At the party we will sell coffee, tea, coca-cola, fanta, sprite, juice, chips, food etc.

At the party we will arrange a demo-competition for both Amiga and C64. The demos will be shown on a large screen, so that you all can get a fair chance to decide which demo to vote for. We expect to have 5000 Dkr. (about 1270 mark or 415 pounds) for each winning demo, but that depends on how many of you are coming (we figure that there are coming at least 500 comp-freaks). Also there will be a second price at 1000 Dkr. (250 mark or 80 pounds). We will have a voting system like the one they use at the 'European song contest'. You will give your vote directly to a computer, and the current results will be displayed on the big screen. This should eliminate any chance of a non-fair competition.

In the time before the competition, there will be non-stop video shown on the big screen. Amok has promised us to bring along their lasershow. Also we will have arcademachines, so we hope that there will be amusements for everybody.

We will hire an electrician to fix the power supply, so that there will be no fuses blown.

For all this we will have to charge you all for 80 Dkr. (about 20 mark or 7 pounds). People without invitations can expect to pay more. You will have to pay with danish kr.

Also you should bring along some valid I.D. (passport, drivers license, etc.).

If you have some friends, who would like to come, you're welcome to invite them along as well.

All major groups will be presented at this party, also a few guys from the states, so show up.

Address:	Upfront! P.O. box 2031 8900 Randers 2 Denmark	Dominators S.J.P. 118 4700 Naestved Denmark	Trilogy Box 57 3480 Fredensborg Denmark
Phone:	(DK)+ 6441491 (Henning)	(DK)+ 3734349 (Danny)	

We would be most pleased, if you would reply this invitation, by contacting us and tell how many of you are coming, and how many computers you expect to bring along.

P.S. When you come near to Randers, you should tune your radio in on 95.3 MHz and listen to our local radio station.

Name of your group:
How many people are coming:
How many Computers:

Figure 3.7 Physical invitation letter for the 1989 Dominators/Upfront/Trilogy Party, emphasizing event professionalism through an electrician on call, food sold, prizes, and presence of major groups.

For this and similar parties at the time, some things could go unsaid. The organizers take for granted that the reader understands the concept of a computer “party” (people meeting with a focus on piracy and/or creating demos), what a “demo” is (a program demonstrating technical, graphical, and music skills), what a “demo competition” is (a competition where

attendees vote on the best demo for different platforms and/or categories), and the concept of the "group" (a group of people working together to crack software and/or make demos, under handles) behind the invitation, which likely is assigned some authority in the hierarchy of the demoscene. The invitation highlights four selling points:

1. Availability of food and drink in nearby shops and at the party itself
2. Demo competitions with cash prizes
3. "An electrician to fix the power supply, so that there will be no fuses blown"
4. Presence of "all major groups"

The first three points distinguish this event from earlier, less professional parties—like the 1988 Jewels/Danish Gold/Dominators/Upfront Party (figure 3.6)—which lacked food and organization. The mention of an electrician highlights the professionalism of the event, and the "major groups" signal the event's importance, as organizer Einstein explained:

> **Einstein:** The thing about electricians in the invitation was because we had been to parties where the power had gone when too much power had been drawn. Then, of course, it should be included in the invitation that the supply of power would be under control.
>
> **Jesper Juul:** Why does it say, "all major groups"? If you had to explain to someone who hadn't been there, why was "all major groups" important for a demoparty?
>
> **Einstein:** It's like if you have to lure people to a football event, it would be cool if Lionel Messi and Ronaldo show up. That is what it is all about. If all the big ones are there, then others will want to come too. It's not just such a small event that some nobodies organize.[37]

The party took place at the Nørrevangskolen public school in Randers, Denmark, from December 26 to 30, 1989. What happened at the party? Figure 3.8 shows participants in their late teens seated at desks with their

computers. The official program describes Amiga and C64 demo competitions and other activities such as movies and arcade games.

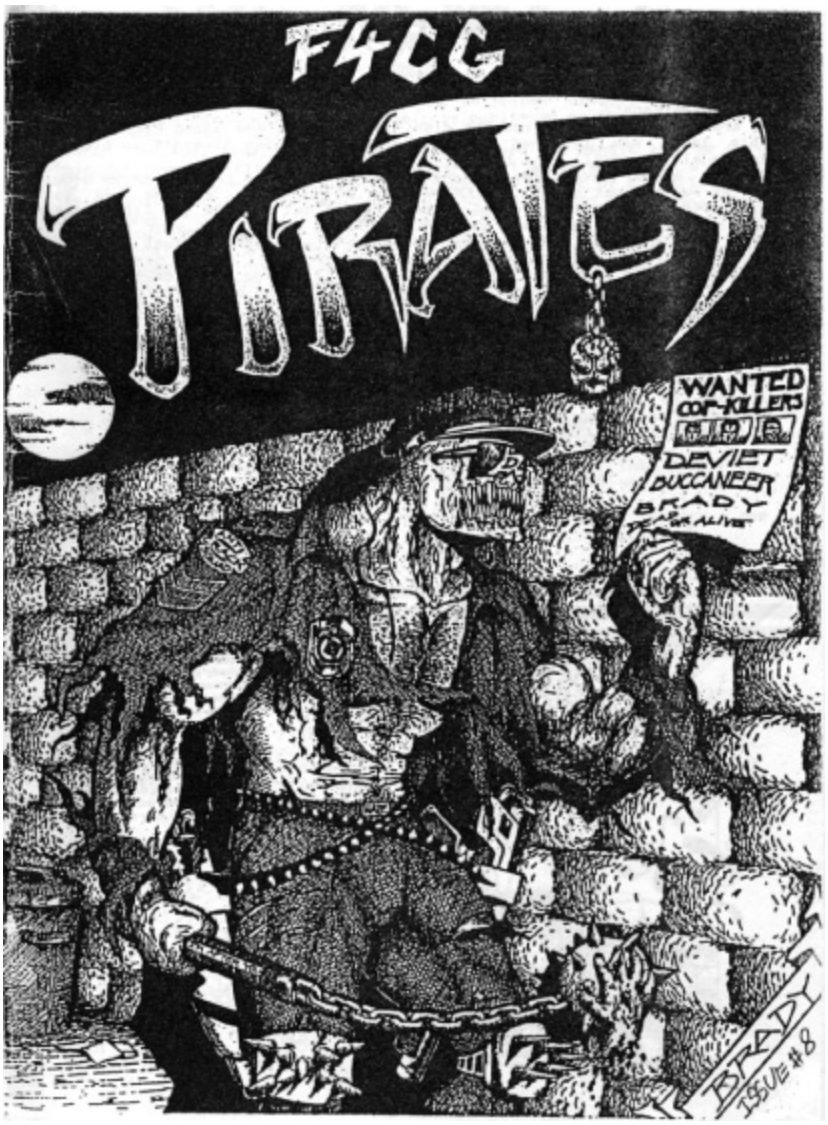
Randers, Denmark, December 26–30, 1989

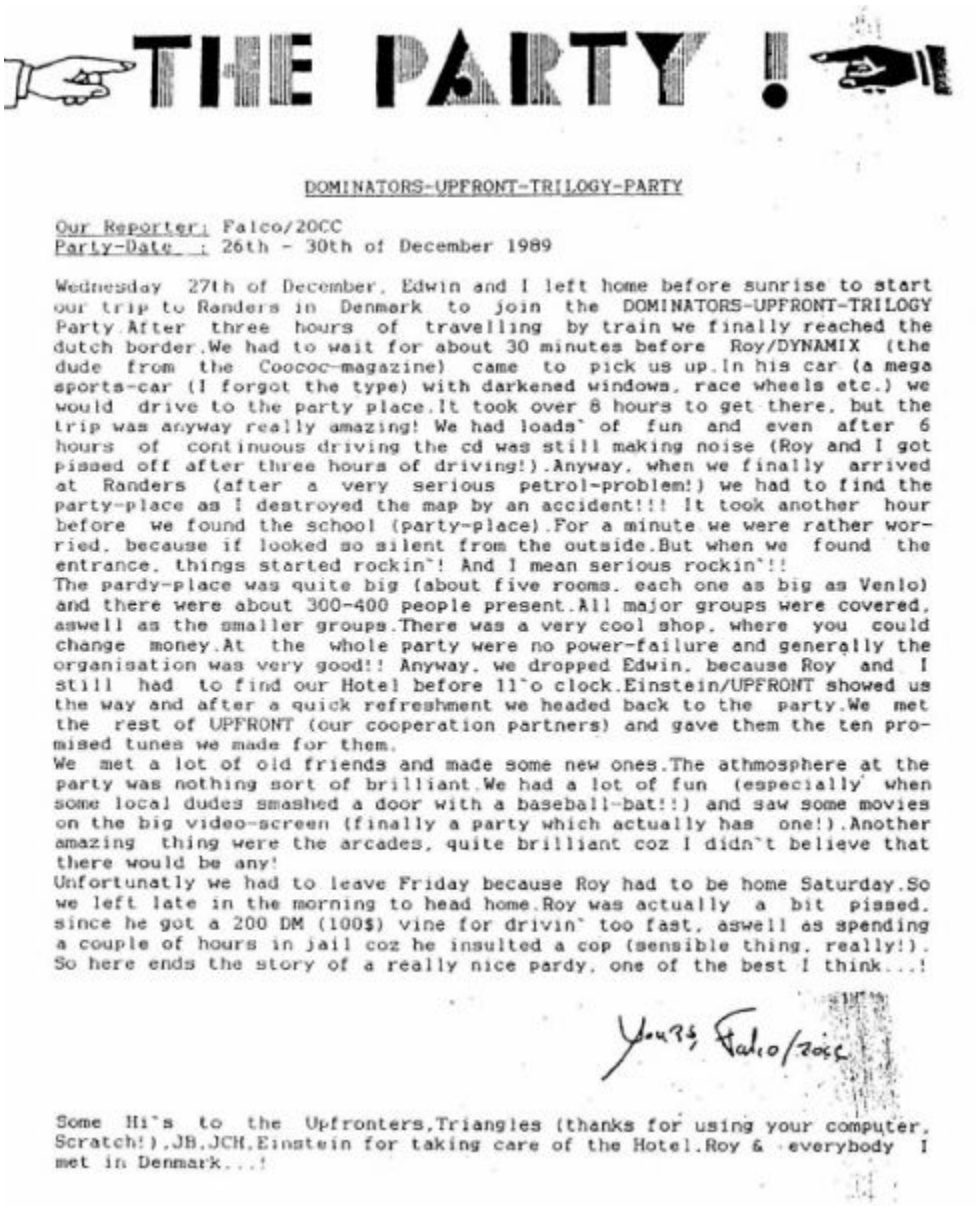
It is the biggest and most important party of 1989, four hundred boys in their late teens and early 20s meeting in a school in the small city of Randers. Being here is important because we want to show how good we are as group, and we have worked for months on our demo [figure 3.11], the biggest and best we have ever made. Being only slightly pretentious and eager to demonstrate my cultural knowledge, I have proposed the demo's name, *The Road of Excess*, from the William Blake quote, "The road of excess leads to the palace of wisdom." My role in the Triangle group is as a programmer, and I am planning four or five demo parts. We talk about making something more designerly than the typical demo, and we want to move beyond the default black background of demos. The demoscene feels a lot like any other gathering of teen boys at the time—some are into pop music, some of us in our Triangle demo group think ourselves alternative and are eager to show it with callouts to our favorite bands, imitating logos from The Doors, Einstürzende Neubauten, Public Image Limited, and indirectly referencing Talking Heads and King Crimson. We also work hard to show technically challenging parts, like the perspective scrolling image in the sideborder, the 3D objects, the scrolling FLI image, the *Breakout* game playable in the sideborder. I am torn between the different quality criteria for demos. I like the technical challenge but am appalled when a younger party participant looks at my screen of moving sprites and reduces its value to a banal question: "How many are there?" Deadlines are meant to be moved, and we work on our demo 'til 2 a.m., my floppy drive becoming so warm that I have to disassemble it and remove its top to continue. We are disappointed with our third place, but the winning demo clearly works better as an audience pleaser, with more clearly signaled technical tricks. We sleep on the hardwood floor of the gym in sleeping bags and drink beer during downtime.

Reports of the Party

We can compare the promotion of the party with party reports, found in disk magazines ("diskmags") and photocopied magazines. *Pirates* magazine (figure 3.9) reported on the professionalism of the event, the size of the event (300 to 400 people), and on all the friends met.[38] The *Coococ* report mirrored the selling points of the invitation, noting the size of the event, the presence of "every major group," and emphasized the social nature of the trip there.[39] Completely different was the lengthy report of diskmag *Fatal News*, which focused on the party aspect and mainly chronicled participants consuming excessive amounts of alcohol, going to local discos, getting kicked out, and almost getting into fights, identifying participants by group and pseudonymous handle. The report mentioned in passing that "there was almost nothing cracked at the party" and complained about the C64 demo competition happening at 2 a.m.[40] *BYG* reported on the important groups represented at the event and mentioned the new games (obviously pirated).[41] The *Lethal News* report (written by aforementioned Trap) declared it "the party of the year," celebrating meeting people but bemoaning blacking out the first day due to alcohol and drug intake and complaining about the late C64 competition.[42]

Figure 3.8 Top: Participants at the 1989 demoparty seated in school halls with their computers. Middle: The author in the blue shirt. Bottom: Party organizer Einstein sleeping under a table and under paper printouts. (Photos by Jan Lund Thomsen.)


F4CG
PIRATES
WANTED
COP-KILLERS
DEVIET
BUCCANEER
BRADY
DEAD OR ALIVE
BRADY
ISSUE #8



THE PARTY !

DOMINATORS-UPFRONT-TRILOGY-PARTY

Our Reporter: Falco/20CC
Party-Date : 26th - 30th of December 1989

Wednesday 27th of December, Edwin and I left home before sunrise to start our trip to Randers in Denmark to join the DOMINATORS-UPFRONT-TRILOGY Party.After three hours of travelling by train we finally reached the dutch border.We had to wait for about 30 minutes before Roy/DYNAMIX (the dude from the Coococ-magazine) came to pick us up.In his car (a mega sports-car (I forgot the type) with darkened windows, race wheels etc.) we would drive to the party place.It took over 8 hours to get there, but the trip was anyway really amazing! We had loads` of fun and even after 6 hours of continuous driving the cd was still making noise (Roy and I got pissed off after three hours of driving!).Anyway, when we finally arrived at Randers (after a very serious petrol-problem!) we had to find the party-place as I destroyed the map by an accident!!! It took another hour before we found the school (party-place).For a minute we were rather worried, because if looked so silent from the outside.But when we found the entrance, things started rockin`! And I mean serious rockin`!!
The pardy-place was quite big (about five rooms, each one as big as Venlo) and there were about 300-400 people present.All major groups were covered, aswell as the smaller groups.There was a very cool shop, where you could change money.At the whole party were no power-failure and generally the organisation was very good!! Anyway, we dropped Edwin, because Roy and I still had to find our Hotel before 11`o clock.Einstein/UPFRONT showed us the way and after a quick refreshment we headed back to the party.We met the rest of UPFRONT (our cooperation partners) and gave them the ten promised tunes we made for them.
We met a lot of old friends and made some new ones.The athmosphere at the party was nothing sort of brilliant.We had a lot of fun (especially when some local dudes smashed a door with a baseball-bat!!) and saw some movies on the big video-screen (finally a party which actually has one!).Another amazing thing were the arcades, quite brilliant coz I didn`t believe that there would be any!
Unfortunatly we had to leave Friday because Roy had to be home Saturday.So we left late in the morning to head home.Roy was actually a bit pissed, since he got a 200 DM (100$) vine for drivin` too fast, aswell as spending a couple of hours in jail coz he insulted a cop (sensible thing, really!).
So here ends the story of a really nice pardy, one of the best I think...!

Yours, Falco/20CC

Some Hi`s to the Upfronters,Triangles (thanks for using your computer, Scratch!),JB,JCH,Einstein for taking care of the Hotel,Roy & everybody I met in Denmark...!

Figure 3.9 *Pirates* 8 (1990). Left: *Pirates* was an underground photocopied magazine about cracking and the demoscene. Right: A report about the 1989 party.

Coverage of the party could also be found in the specialist press, with Danish *Nye Computer* magazine excitedly explaining the history and concept of demos to readers and praising the C64 and Amiga demos.[43] Counterintuitively, the print magazine went into more detail with technical aspects of the demos than did the demoscene reports. The article presented

the party as exemplary for demoparties and discussed the winning demo enthusiastically.

The Aesthetics of 1989 C64 Demos

What role did demos play at the demoparty? Groups could submit demos to the announced competition, demos were shown on a projector at 2 a.m., and attendees voted for their favorite demo by awarding points on voting sheets. According to the official C64 demo results, the group Bones' *The Larch 3* took first place, Upfront's demo came in second place, and Triangle's *The Road of Excess* (my group's demo) came in third.[44] The Upfront demo was regrettably lost to time, but we can compare the other two demos. (► See demos on the website.)

Bones' winning entry *Larch 3* (figure 3.10) is oriented towards technical effects with nearly every screen claiming technical novelty. The *sideborder multiplexer* (more than eight sprites in the border) is preceded by a screen explaining why the effect is technically impressive. The same goes for the *raster split* (changing background color multiple times to divide the screen in columns), the vertical *sprite stretching* routine, and real-time 3D with hidden lines. The demo's focus is on what is considered difficult or impossible.

Compare this to Triangle's third-place *Road of Excess* (figure 3.11) whose technical claims are less highlighted than in the Bones demo. *Road of Excess* is more designed and more centered on cultural references—both in the title's reference to William Blake's poetry and in callouts to bands. The second screen reuses the now well-known Tutankhamen image but remakes it in the FLI mode (discussed later) with additional colors and scrolling the image. The third screen first scrolls in the Triangle logo in perspective (and sideborder), then displays 3D objects. One screen features a playable version of *Breakout*, this time in the sideborder. This demo is also making technical claims, but it makes additional claims for cultural status.

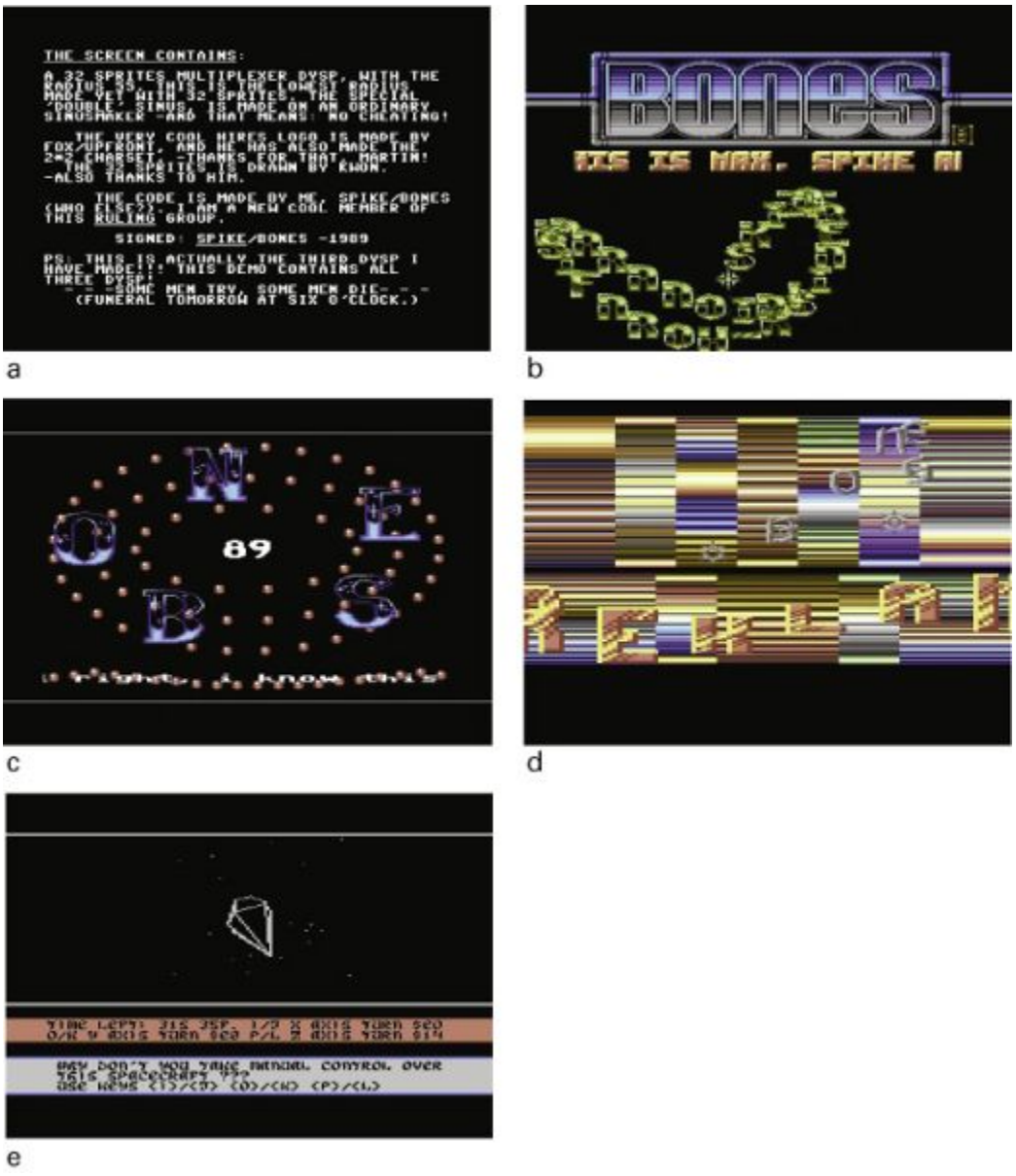


Figure 3.10 *The Larch 3* demo (Bones, 1989) emphasizes its technical achievements in text. a) Explanation of technical claims for a screen; b) sideborder multiplexer; c) software sprites; d) raster splits; e) real-time 3D.

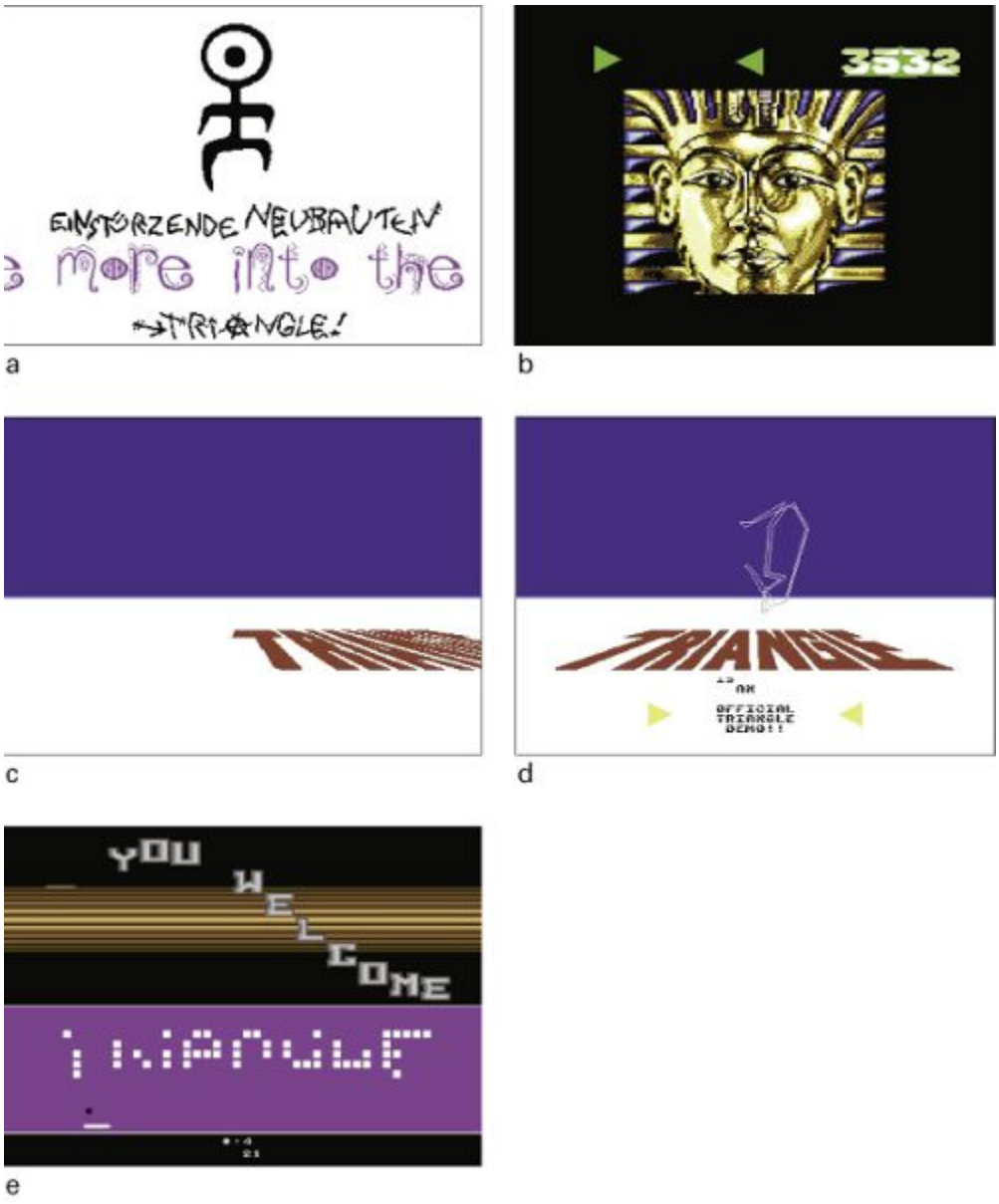


Figure 3.11 *The Road of Excess* demo (Triangle, 1989), less explicit in technical claims but references graphic design and makes claims for cultural knowledge in the William Blake reference of the title and inside the demo. a) Reference to cultural idols; b) scrolling the Tutankhamen image with FLI technique; c) perspective scrolling of logo in sideborder; d) 3D model of glasses; e) the game *Breakout* playable in the sideborder.

Are these good demos? Clearly, *Larch 3* was considered the better demo. Lassi Tasajärvi argues, “The point of a demo is to make the viewer speechless. After a period of speechlessness, the question that comes to

mind ought to be ‘How IS that done?’”[45] I agree that this is a central point of early demos, but not the only one. It has been common to divide demo history into several “schools” with different characteristics.[46] Tasajärvi’s 2004 book divides the history of demos into three:[47]

- **Oldskool: The Classic Era** (1980–1991), with programming-oriented demos.
- **Middleskool: The Modern Era** (1992–1996), with a focus on design and art.
- **Nuskool: The Post-Modern Era** (1997–) with music video–like demos.

Obviously, culture like demos rarely falls neatly into such periods.[48] I suggest we think about it in a different way because it is, as I mentioned, not obvious why demos exist. A word processor has the function of writing text, and a game generally provides entertainment, so a demo must argue for its value, content, and design choices. *Why is a demo important, and who is it for?*

Where early demos were organized as a series of screens each showing an effect, where the viewer had to press a key to progress from a screen, many later demos are structured more like music videos (*trackmo* or “Nuskool” in Tasajärvi’s terms) of a fixed duration of a few minutes. Such demos are designed to entertain the audience by being structured around a graphical composition, story, or theme. This kind of demo also responds to the format of demo competitions, which often involve tired demoparty participants standing while watching demos. Some early demos took fifteen or twenty minutes to go through, so shorter and more focused demos were greatly appreciated by the audience waiting to vote on their favorite demo.

The Technical Trick Enabling the Design, and the Design Hiding the Technical Limitation

I do not mean to say that demo design is a choice between technical skill and other kinds of value. Rather, successful demos not only combine several such claims of importance, but all but the most technically oriented demos are based on the combination of technical tricks and a design that is

comprehensible to nontechnical viewers. In the late 1980s, it was common to make demos showing more than eight sprites either as “sine scrollers” of characters moving in waves or by moving the sprites in a circle (figure 3.10). Moving sprites in circles or waves are technical tricks that appear natural. Without counting, we can see that there are many more than eight sprites on screen, and we get the impression that they can move about freely. Moving many sprites in a circular pattern works technically because a circle distributes sprites vertically on the screen, meaning that there never needs to be more than eight sprites on a horizontal line, thus making it work with the VIC-II chip’s limitations.[49] Making such demos therefore requires not just technical insight but the identification of a natural-seeming pattern that viewers intuitively understand. Demos thus generally have designs that appear like a natural fit to the technical trick, such that what happens is understandable by both a lay audience that may be impressed by the sheer amount of movement, and by a technical audience that can parse the technical claims being made. *The technical trick makes the design possible, and the design makes it feel as if the technical limitation isn’t there*.

The Social Scene

Stereotypes of programmers often present them as people singularly focused on *things*. It is striking then to consider the amount of effort that went into maintaining social status and connections through demo effects and greetings in demos. Much of the material from the 1989 Randers party —physical zines, diskmags, and in-program scrolling text—can read like the society page of a magazine, with intense care taken to mentioning relevant groups, reporting on events, and general social maneuvering. Like other magazines, *Pirates* had regular updates about social happenings in the demoscene. For example, *Pirates* issue 8 featured two full “news” pages dedicated exclusively to reporting who had entered or left which groups, in addition to a rumors page, a dedicated Swedish gossip page, and a report on the Finnish scene.[50] *Coococ* issue 10 also featured gossip about scene members.[51]

At this time, demos usually featured long “greeting” sections, naming other groups to gain and maintain status. Einstein recalled social

maneuvering and interactions as a central preoccupation, where he made many friends by writing (physical) letters:

> The social aspect was really important. We met in the group. If you are a coder [programmer] on a daily basis, it is a loner job. But I had friends all over the world, I had one in Norway, I corresponded a lot with some people in the Netherlands, and there are some I have never met. When you sent a floppy disk, you also wrote a letter to the person, that was quite natural. I don't remember what was written in those letters, but that was part of it.
>
> The greetings were to show that you knew a lot of the other important players.[52]

The early demoscene was a social environment characterized by an intense desire for validation and status. I do not mean this in the banal sense—that we may think the demoscene seems technical but "actually" people are socializing around an activity—but that in the examined material, social interactions play a surprisingly large role, such that reports on friends, partying, and group changes take up vastly more space than any technical discussion of the demos.

Swyx described the desire for status and validation as central to both the demoscene at large and to the making of the culture's magazines:

> It dawned on me that beyond just being fun, having ambition, and wanting something, we were actually driven by what others thought of us. This was emphasized by charts for everything—best demo groups, best cracking groups, best everything. And that was some of the most important stuff for me. Every time an *Illegal!* came out, which I had typically contributed to, I would start by looking at those ridiculous charts on the back.[53]

Both Swyx and Einstein independently proposed that competitions and charts were to be expected, and both invoked sports as a comparison.

> **Swyx:** Well, isn't it natural? Otherwise, what we could have done at that time would have been to play soccer or team handball, or something where we compete for things. I think we had to have some competition around it.[54]

Of course, just as there were different ideas about what made a good demo, different participants had different motivations. Among interviewees, Chix described his own motivation as being mostly about solving technical challenges, rather than about competition, even if demoscene events were organized around social interaction.[55]

The early demoscene was almost exclusively male, from group members to party organizers to those staffing the food stalls at parties. Hege Nordli has explored the gender skew of computer parties and examined newer strategies for inclusion at LAN parties, which have led to parties being less male-dominated.[56] While my interviewees agreed that the demoscene was overwhelmingly male, Swyx also noted that it was "unsexy": "It was nerdy. Totally unsexy. Everyone agreed on that, everyone knew it. . . . Not because of the activity itself, but because of the choice to spend so much time sitting and hacking away on a computer instead of doing all sorts of other things."[57] The gender skew was not considered surprising. As we saw in chapter 1, Commodore had already decided on gender roles for the C64 by the 1982 launch. It advertised the C64 with young men programming and playing video games, while girls could play educational games and grown women could attend to secretarial or homemaking work. There was wide cultural agreement that computer programming, cracking, and demos were for boys and young men, and there was little recognition that it could be any different.

Programming Demos

The border tricks were entirely unplanned by the C64 designers. In 2022, C128Neo circuit board designer Johan Grip showed a modern demo to chip designer Al Charpentier:

> We showed him [Al Charpentier] *Edge of Disgrace*, one of the seminal C64 demos, and he was amazed. This is the man who designed the

> VIC-II chip, so I ended up explaining how to remove the top/bottom border. I explained how you wait to raster line to 248, and then you flip the register. And he goes, “but it wouldn’t draw anything.” So I said, “it would keep drawing sprites.” Then he got quiet for a couple of seconds, and you could see he was thinking. And then he goes, “yes, yes, it would.” I could almost see that he was looking at the schematics of the VIC-II and going, “Yep, it will draw sprites.”[58]

When I interviewed him, Al Charpentier was totally accepting of the ways developers had found to exploit his design, did not feel subverted, and had nothing but admiration for the tricks. Why? It helps to understand that in 1981 when the C64 was being designed, lessons from the Atari 2600 had been absorbed, and it had become clear that it was a good thing when developers used a limited system to create a wide range of unforeseen effects, and that you could design hardware with this in mind. Charpentier described flexibility for the developers as a goal for hardware design, “if the hardware is amenable to different solutions, it opens up a lot more horizons.”[59] If we in the demoscene considered ourselves to be subversive or working against intentions, that wasn’t always felt by the people whose intentions we imagined ourselves up against.

The border tricks removed the border, but why was there a border in the C64 image in the first place? The 27-inch monitor I am typing this on shows pixels to the edge of the physical screen with no border. The C64 border exists for two reasons. First, the C64 was designed to connect to analog TVs that show only an unpredictable subset of the image from the computer (or television station), and the border thus serves as margin to make sure the central text area is visible on all TVs. Second, the limited speed of C64 memory means that there must be a border if there are to be sprites, as the VIC-II chip works just at the limit of what is possible with the amount and speed of the available RAM.[60]

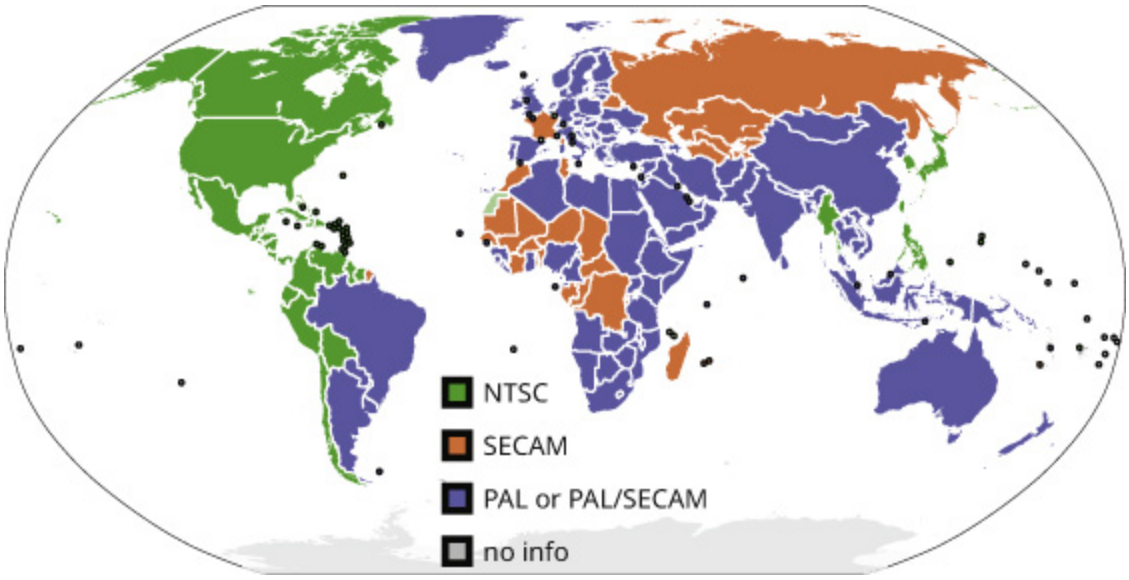


Figure 3.12 World map of analog television standards, showing the world divided according to colonial and later geopolitical history. Software developed too specifically for North American NTSC C64s does not run on the European PAL C64s, and vice versa. This divided much of C64 culture. (Illustration by Akomor1, public domain, via Wikimedia Commons.)

There were two main competing analog TV standards, NTSC (North America, South America, Japan) and PAL (most of Europe, Africa, Oceania, and Asia), and therefore two different C64 models. Figure 3.12 shows their global distribution, which followed colonial and geopolitical history. (French SECAM and South American PAL-N/M are like PAL, except they only show the standard C64 image in black and white.) NTSC shows an image 60 times per second (Hz), while PAL operates at 50 Hz. The PAL C64 image has a larger border, giving more time for programmers to do effects and calculations. This difference between TV standards meant that more advanced programs developed for one standard would not work on the other, and especially the European demoscene was inaccessible to Americans until the arrival of good emulators. For games, crackers did some of the work of "fixing" NTSC games to run on PAL machines and vice versa.[61]

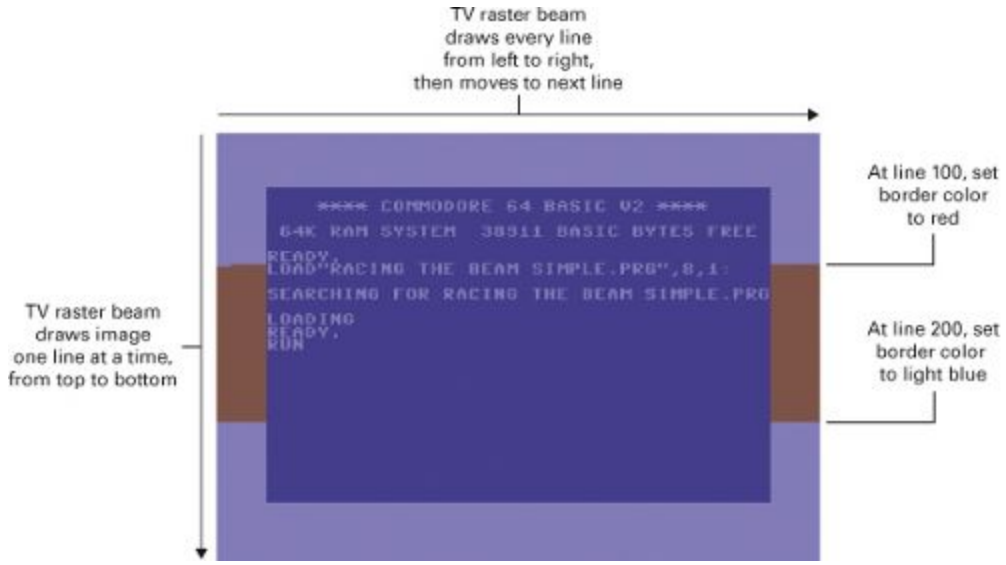


Figure 3.13 The TV's raster beam draws the image line by line from top to bottom. Here a small program waits for the raster beam to reach line 100, then sets the border color to red; it waits for line 200, then sets the color back to light blue. No character, bitmaps, or sprites are changed, just the border color.

Let me explain some of the more common demo effects for the C64, how they result from the design of the machine, and what they meant for the demoscene.

Raster Beams and Raster Interrupts

To understand demos, we should first know that when programming graphics on the C64 and other game-oriented systems of the era, you need to always consider the current position of the raster beam of the CRT television. The raster beam is the concrete beam of light on the TV screen that draws the TV image one line at a time from left to right and top to bottom. On the Atari 2600, programming in direct relation to the raster beam was called *racing the beam*.[62] In the 1980s, you could tell a game-oriented system from a "serious" one by the presence of facilities to synchronize your program with the raster beam, as serious machines were apparently above such considerations. As an example of working with the raster beam on the C64, figure 3.13 shows a simple program that waits for lines 100 and 200 and changes the color of the border.

Larger programs usually use the raster interrupt technique, specifying a piece of program to be executed when the raster beam reaches a given line. (► See programs and source code on the website.)

Changing the Border Color by Waiting for the Raster Beam

```
MAIN
        SEI
LOOP
        LDA #100       ;WAIT FOR LINE 100
WAIT2   CMP $D012
        BNE WAIT2
```

	LDA #2	;RED
	STA $D020	
	LDA #200	;WAIT FOR LINE 200
WAIT3	CMP $D012	
	BNE WAIT3	
	LDA #14	;LIGHT BLUE AGAIN
	STA $D020	
	JMP LOOP	

Sprite Multiplexing

The demoscene had invented the border tricks, but we also thought we had invented showing more than eight sprites on the screen at a time. Yet, going back to the original C64 documentation, the *Programmer's Reference Guide* mentions that more sprites can be displayed. I have no recollection of reading this statement at the time, but it really is in the book, "Up to 8 sprites at a time can be maintained by the VIC-II chip automatically. More sprites can be displayed using RASTER INTERRUPT techniques. . . . These special sprite abilities make it simple to program many arcade style games."[63]

The VIC-II chip can show exactly eight sprites on its own. Other graphics chips of the time, like the TI-99/4A chip and the later Nintendo Entertainment System (NES) chip, also had limitations about how many sprites could be shown on a horizontal line but handled their reuse —"multiplexing"—for the programmer. Because this was left to the programmer on the C64, most early games would only show eight sprites, and as a result, the C64 appeared to gain abilities as programmers learned to show more.

Figure 3.14 shows a minimal sprite multiplexing routine. The program waits for the TV's raster beam to reach line 10 at the top of the screen, and then places the first eight sprites at line 80. After the raster beam has moved down and displayed these sprites, the program moves the sprites to line 180 in the bottom half of the screen, where they are displayed once more as the raster beam draws them there. We repeat this, and the simple program shows sixteen sprites in total, like in the *Summer Games* opening ceremony (figure 2.13).

Figure 3.14 How can we show more than eight sprites? This simple "multiplexer" divides the screen into two sections. Once the first eight sprites have been shown, they are moved down and are shown once more.

Core Program Showing Eight Sprites in Two Places on the Screen

```
          SEI
      ;WAIT FOR LINE 10
LOOP       LDA #10
LOOPB     CMP $D012
            BNE LOOPB

      ;PLACE SPRITES IN THE TOP OF THE SCREEN
            LDY #14
            LDA #80
VPOS1     STA $D001,Y
            DEY
            DEY
            BPL VPOS1

      ;WAIT FOR LINE 128
            LDA #128
```

```
LOOPMB    CMP $D012
            BNE LOOPMB

    ;PLACE SPRITES IN THE BOTTOM OF THE SCREEN
            LDY #14
            LDA #180
VPOS2     STA $D001,Y
            DEY
            DEY
            BPL VPOS2
            JMP LOOP
```

Around 1987, demos began to show larger number of sprites moving freely, and competition for showing the most sprites grew, scroll texts boasting about the accomplishments. How would you, for example, show thirty-two sprites, seemingly moving freely?

Figure 3.15 shows one such sprite multiplexer program that I wrote in 1987, displaying the Commodore balloon thirty-two times. The colors in the border show the time taken every frame by raster interrupts when new sprite positions are calculated (orange), the sprite list is sorted (red), the first sprites are positioned (black), and when sprites are moved for reuse (gray).[64] (► See programs and source code on the website.)

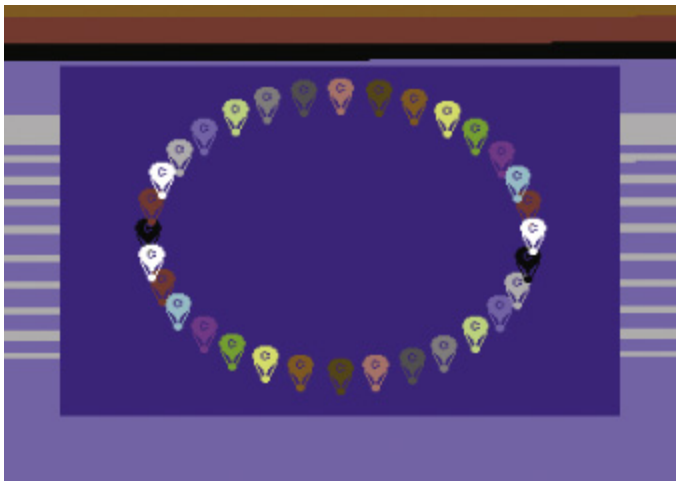


Figure 3.15 Sprite multiplexer showing thirty-two sprites (seemingly) moving freely. Colored bars on the side show the time the program spends placing the sprites.

Opening the Borders

Having seen how sprite multiplexing works with the position of the raster beam, it becomes easier to understand the border tricks in the demos shown at the beginning of the chapter (figure 3.1). How can programs hide the border? It all stems from a glitch in the scrolling facilities of the C64. To prevent the new lines from suddenly appearing when we scroll the screen, the VIC-II allows us to expand the border to cover more of the character area. The VIC-II patiently waits for the correct raster beam position before starting the border, but if you switch the size of the border at the right time, the VIC-II will now be waiting for a raster beam position that has already passed, and the chip will never begin to show the border. The sideborder trick of opening the border left and right works by the same principle but requires more precise programming.[65] Because the border is just the top layer of the C64 display, removing the border reveals sprites and background color underneath (figure 3.16). (► See programs and source code on the website.)

Figure 3.16 Switching to 24-line mode just after the expanded 24-line border would have started makes the VIC-II chip forget to draw a border, allowing sprites to be shown.

Opening the Top/Bottom Borders

To open the top/bottom border, wait for the bottom of the text area, enable 24-line mode, then wait for the top of the screen and enable 25-line mode.

```
          LDA #0
      ;CLEAR GARBAGE LINES or “ghost bytes”
              STA $3FFF
              SEI
      ;WAIT FOR BOTTOM OF SCREEN
BOTTOM    LDA #$F9
BOTTM1    CMP $D012
              BNE BOTTM1

      ;24-LINE MODE, MAKING VIC-II FORGET TO SHOW
  BORDER
              LDA #$13
```

```
              STA $D011

    ;WAIT FOR TOP OF SCREEN
ZERO      LDA $D011
              BPL ZERO
ZERO2     LDA $D011
              BMI ZERO2

    ;BACK TO 25 LINES
              LDA #$1B
              STA $D011
              JMP BOTTOM
```

More Fun with the Scroll Registers

All of this created high hopes for the future, and programmers would readily claim to have created new tricks they could name. Some of these allowed the C64 to do new expressive things, others were names for obvious combinations of existing tricks, and some were only meaningful to those with a deep technical knowledge of the machine. Multiplexing and border tricks responded to real limitations of the C64 and their effects were obvious even to the uninitiated, but later tricks became more subtle. Timing became a general measure of things. Could you do a multiplexer in the sideborder while maintaining the required timing? While also displaying characters (on “badlines”)?[66] Once it had become clear that effects could come from manipulating the VIC-II chip in nonobvious ways, new tricks were discovered. To scroll the screen vertically, you change the screen’s vertical position once per frame. But what if you manipulate the register more often? This became the last three truly useful effects, *FLD* (*Flexible Line Distance*), *FLI* (*Flexible Line Interpretation*), and *VSP* (*Variable Screen Placement*).

Figure 3.17 The *FLD* (*Flexible Line Distance*) effect allows for space between lines; *FLI* (*Flexible Line Interpretation*) is the reverse technique, allowing more colors to be shown in a bitmap image (*So-Phisticated III*, 20th Century Composers, Black Mail, 1989).

In FLD (figure 3.17, left), we can trick the VIC-II chip much like in the border tricks. If we add one to the vertical scrolling position every line, we can keep the VIC-II chip from realizing that a new line should be shown. This allows us to create distance between lines, and we can also move an entire picture in smoothly without having to update character or bitmap memory. (The opposite effect is “linecrunch,” which skips lines.) Again, such an effect is impressive if you understand the regular limitations of the machine and know that this effect goes beyond what is considered possible.

Conversely, FLI (figure 3.17, right) forces the VIC-II to think that every pixel line is a new row of characters. Why would that be important? Because the bitmap modes use character input to determine which colors are available in each 8-by-8-character cell. By forcing the VIC-II chip to reread the characters, more colors can be shown in a bitmap image. This effect can be subtle, and graphics artists often went out of their way to create images with gradients that wouldn’t otherwise have been possible. Finally, VSP (Variable Screen Placement) allows us to move the entire screen horizontally instead, but the technique is famous for crashing some C64s, the mechanism of which was not fully understood until 2013.[67]

Figure 3.18 *Edge of Disgrace* (Booze Design, 2008) is a later demo that highlights algorithmic visual effects and makes interesting screen compositions including using novel color combinations.

Though exploitable glitches in the VIC-II became further between, there continued to be a race for naming new techniques. The C64 wiki lists no fewer than thirty such new "graphics modes" with abbreviations, names, and dates of creation, including "AFLI—Advanced Flexible Line Interpretation, April 1990," "AH—Advanced Hi-res, December 1996," and "NUFLI—New Underlayed Flexible Line Interpretation, July 2009."[68] Some of these have limited use, while the more popular like NUFLI are supported by utilities for converting, editing, and manipulating images. In all cases, we see that the technical culture of the demoscene has a strong competitive component of claiming firsts and thereby status.

Later Demos and the Puzzling C64

While a full history of the C64 demoscene is outside the scope of this book, it is worthwhile to understand how the demoscene later developed. (► See demos on the website.) Figure 3.18 shows a later demo, the 2008 *Edge of Disgrace*, where the C64 is used to produce graphically interesting visual effects with surprising colors, laying out the screen and appearing more designed than the previous demos. Demos of this type rarely claim to contain novel hardware effects, but they will present puzzles for the viewer, as much effort goes into breaking free from standard C64 dimensions (such as the size of a sprite), layout (such as the border), and 90-degree angles.

These demos do often numerically improve on existing techniques or combine techniques such that it is nonobvious what is happening technically.

Figure 3.19 The recent *Memento Mori* demo (Genesis Project, 2020) contains nonobvious effects, such as double-layered scrolling. This is packaged with references to art history, including to William Blake.

Finally, figure 3.19 shows the 2020 *Memento Mori* demo, named for the artistic motif of remembering death. The demo combines technically challenging effects such as sideborder scrolls and graphical rotation, as well as double-layered scrolling which poses a puzzle to the viewer, all the while demonstrating cultural knowledge through references to art history and to William Blake's poem "Ah! Sun-flower."

To this day, the demoscene has a tension between evaluating demos on technical merits and on their visual, sound, or entertainment value. According to programmer Trap who has participated in the demoscene since the 1980s, "Today the scene is divided between people from the old school who believe that a demo has no justification for existence if there is no code making other programmers wonder how it was done. Then there is the other side, which I am on, which believes that there should be a predominant element of entertainment, a story, something that is fun to watch."[69]

The Puzzle of Demos

How can a new demo present a puzzle when the machine has been explored for forty years? This stems in part from the sheer number of ways a C64 can show pixels on the screen. For example, if you want to create a 2-pixel-high horizontal colored line on the screen (figure 3.20), you can:

1. Make a bitmap screen and draw a horizontal line.
2. Use the regular character set to draw a horizontal line by repeating the same character forty times.
3. Create your own character set and do the same.
4. Create your own character set and make forty different but similar-looking characters, each with a horizontal line.
5. Use a raster interrupt, change the background color, wait two lines, and change it back.
6. Place horizontally expanded sprites with a 2-pixel-high line across the screen.
7. Place horizontally and vertically expanded sprites with a 1-pixel-high line, expanded to two lines, across the screen.
8. Have two bitmap screens, one black and one white, and switch between them so that the colored one is shown for two lines.
9. Change between two character sets for two lines.
10. Use a multicolor character set and change the $D022 character color register for two lines.
11. Draw two full 8-by-8 character lines across the screen but use linecrunch to shrink each to a 1-pixel horizontal line.
12. Create a space between two rows of black characters using FLD (figure 3.17).
13. Set background color to white, use FLD to push entire screen down, and create a horizontal line by only setting $3fff to zero for two lines.
14. Set background color to white, disable characters, and create a horizontal line by only setting $3fff to zero for two lines.
15. Make an FLI picture (figure 3.17) showing two horizontal lines.

You can use many combinations of these techniques. Sometimes a visually singular object can be composed from different C64 facilities. (The line in the example was made with technique 12, FLD.)

Figure 3.20 C64 demos can provide a puzzle in part because C64 graphics can be produced in many ways. There are at least fifteen ways to draw this line.

Demos, Games, and the Government

The 1989 party marked a transition towards professionalization and opening to a broader public. This was tied to getting rid of the connection to illegal software activities.

> **Jesper Juul:** Is it intentional that the invitation says nothing about copyparty? Is it a question of distancing from piracy?
>
> **Einstein:** We had a group called Triton Technology where we cracked games, and we made lots of copies of games and everything. And the reason we founded Upfront, or one of the reasons, was to get away from piracy. That Upfront should be a legal group. I think that's part of the reason why we didn't write copyparty. We didn't want to signal that it was illegal in any way.[70]

It is also notable that the alcohol-oriented party aspect of the demoparty was toned down as events became larger. By the next big party in Denmark, the 1990 Dexion X-Mas Conference, the invitation stated that "due to problems at other meetings, alcoholic drinks are not allowed in the halls."[71] This time, a national newspaper covered the event, declaring it (imprecisely) to be the first such event "open to the public" because "computer freaks are no longer just illegally copying software."[72] We can also consider the 1989 party as the second in a fourteen-year tradition of holiday demoparties in Denmark, running from 1988 to the last, The Party in 2002. The continuation of the C64 demoscene also shows a remarkable consistency, like the Dutch X party, the first of which was in 1995, the most recent being X'2024, twenty-nine years later.

However, reaching a wider audience led to conflicts between the demoscene and game culture. An attendee at The Party 2002 complained that demoparties had become LAN parties, with game-playing erasing the demoscene:

> This time the party went totally gamer, and away from the scene. . . . It used to be a great party with a great atmosphere (lots of demos, sceners, technorave, and lots of people talking to each other n' doing cool stuff, instead of just sitting in front of their PC and playing games). This year it seems like the party has finally admitted what it has almost been the last time—just a big LAN-party.[73]

Similar trends can be seen with other big parties, such as Norway's The Gathering or Sweden's DreamHack, which grew from the demoscene but are now primarily gaming and e-sports events.

I have focused on one specific 1989 event. From this vantage point, the early C64 demoscene is best characterized as a self-conscious culture, torn between making demos that only insiders can understand and reaching a broader public, giving a surprising amount of attention to self-presentation (even in self-naming as a "scene"), social status, gossip, and jockeying for positions in a hierarchy. Most participants were likely middle class with the ability to afford computers and travel and with spare time not already accounted for by work or homework.

In my previous book *Handmade Pixels*, I studied the history of indie video games and found a similar push towards creating institutions such as game festivals that could promote and celebrate "the best" independent games. Unlike the demoscene, though, the first Independent Game Festival (1999) was commercially oriented, with the express goal of connecting smaller developers to publishers.[74] Though some literature sees the demoscene as progenitor to especially the Finnish tech industry, the demoscene appears strictly noncommercial (you cannot make a living from demos).

If we connect the demoscene to the idea of hacking, we will first read the demoscene as subversive and oppositional, using computers in ways not intended and tied to the adjacent activity of cracking software. But 1989 party organizer Einstein explained that to be able to rent the public school for the party, they needed to formally register a civic noncommercial association with members and bylaws, and the resulting association "Youth of the Future" rented the school. At the time, Einstein felt they were exploiting a loophole, but as an adult he understands that they were using the system as intended, "We really thought we were cheating the system, but since then I've learned that it's part of a welfare society that you are allowed to create such associations."[75]

Einstein's observation about the public support for the 1989 party leads us back to the population-adjusted figures for C64 crack and demoscene activity (figure 3.5), which correlate somewhat with richer countries, but also with countries with large public sectors and a culture of voluntary associations.[76] Counterintuitively, some of the early demoscene could simultaneously present itself as subversive, rely on the public sector, and be understood—like Swyx and Einstein argued—as a spare time activity parallel to "regular" activities such as sports.

The Game Magazine That Became Technical

Who was the broader audience for C64 demos though? Researcher and musician Anders Carlsson claims that "demoscene artifacts never reached (far) outside the scene, on the one hand because the wider public was not interested, and on the other hand because generally demosceners did not try

to reach an outsider public."[77] The initial example of the *ESCOS* demo certainly goes against that claim.

We can approach this through C64 magazines of the 1980s and early 1990s. We primarily examined *Zzap!64* (1985–1992/1994), US Commodore computer magazine *Compute!'s Gazette* (1983–1994), and German computer magazine *64'er Magazin* (1984–1994). Due to it being the longest-running game-oriented C64 magazine, we read all 107 issues of *Zzap!64* and its successor *Commodore Force*, looking for articles that either described programming or referenced the technical culture of demos.[78]

I began the previous chapter with *Zzap!64*'s 1985 declaration of independence from programming and other platforms, yet the magazine quickly started covering technical issues.[79] By 1986, the magazine was both writing about demos (from the UK online service CompuNET) and reviewing hardware like Final Cartridge and the Laser Genius assembler for programming C64 machine language.[80] Demoscene tricks appeared in the *Zzap!64* August 1987 column discussing "Transporting Sprites Across the Border," helpfully explaining how to open the top/bottom border in technical language ("clear bit 4 of $D011 at raster position $F9").[81] *Compute!'s Gazette* published a similar program in September of the same year, but, demonstrating the technical orientation of German C64 culture, *64'er Magazin* had already published a "Hyperscreen" program with the effect nearly two years earlier in December of 1985, complete with machine language listing.[82]

The following years saw an on/off relationship with the demoscene for *Zzap!64*, sometimes covering demos, interspersed with technically oriented articles.[83] Beyond its dedicated coverage of technical topics, the magazine was inconsistent in what it concretely covered, but we must assume that the contents of a magazine is connected to changes in its reader base.[84] *Zzap!64* discussed demos and programming, so part of the game-oriented readership must have been interested in demoscene culture and programming.

In the 1990s, German *64'er Magazin* reported continuously on demos, explaining demoscene tricks in detail with machine language programming in the Assembler Corner column.[85] The news section told readers how to sign up for The Party in Denmark in 1993, reported on installments of The

Party, and created its own charts for graphics artists, musicians, coders, demo groups, and disk mags.[86]

This certainly contradicts Carlsson's picture of an insular demoscene, but every country was different. US *Compute's Gazette* retained a focus on programming throughout its history and mentioned opening the borders but did not cover the demoscene. The Italian version of *Zzap!64* neither covered the demoscene nor programming. I asked Matteo Bittanti, early *Zzap! Italy* journalist about the reason for this difference, and he explained that Italian C64 culture was not very technical, but that there were "a few specialized publications that catered specifically to providing such instructional content, albeit with minimal reception and limited appeal among the gaming community, if compared to the UK, France or Germany."[87]

As discussed in chapter 2, Graeme Kirkpatrick argued that during the 1980s, games ceased to be seen as software, and magazines developed new terms such as *gameplay* to talk about games in nontechnical language. Kirkpatrick acknowledges that game culture is defined by several unresolved "constitutive tensions," including in relation to technology.[88] We have seen an opposite trend here, where programming and technical subjects became not separated but increasingly integrated into game culture from 1985 to 1994, and magazines as a whole increasingly presented a C64 game culture where programming was integral and where demos were promoted as important.

The Transistor and the Scene

In her study of at-home game creation, Melanie Swalwell notes that when users program themselves, it is not a trivial footnote but rather should be considered a significant part of video game and computer history.[89] The demoscene had considerable reach in Europe, creating an imaginary of the C64 as a ticket to an interesting, cutting-edge culture.

The technical and social aspects of the demoscene were completely intertwined. If Al Charpentier had developed the VIC-II chip slightly differently, it might only have read the 38/40 column, 24/25 line, and scroll registers at the top of the frame, and hence all the central tricks would have been impossible. The border tricks were found by trial and error in the

1980s, but by the 2010s, photos and documentation of the VIC-II chip had become so advanced that the trick had been analyzed in the physical chip layout (figure 3.21), and it was possible to identify the individual chip components responsible for activating (and not activating) the border.[90] Creating standalone graphical programs (demos) became popular in part due to the then-new border tricks; 1001 Crew wanted to demonstrate their new tricks and felt a crack intro was too limited. The VIC-II's numerous possibilities for constructing graphics gave flexibility to developers and allowed demos to work as puzzles, as viewers had to consider the myriad ways an effect could have been constructed. The demoscene, pushed by small components in the VIC-II chip, took in the idea of the party (parties where alcohol is consumed), the idea of groups from music (perhaps hip-hop) or graffiti taggers, and the assumption that rivalry should be supported by formalized competition and rankings but judged by peers rather than by external authorities. The effects made it possible to create programs that could do things that no one had thought possible and created the expectation that the C64 could continue to do more. There might have been a demoscene anyway, but the early demoscene that was, with all its gossip and social maneuvering, appeared concretely in collaboration with the VIC-II chip's transistors, regardless of whether the programmed effects were intended by the machine's designers.

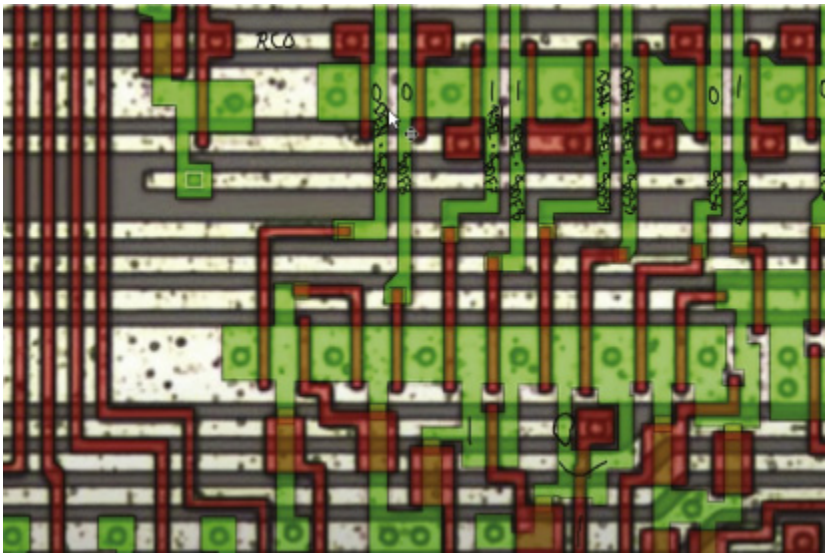


Figure 3.21 The Demoscene Starts Here. Magnification and analysis of the VIC-II chip, pointing to the chip components that can be tricked into failing to activate the border (Bwack 2016).

4

Keeping Up with the Future

Imaginary #4:	It may be old, but the Commodore 64 can still do it!
Programming Language:	Machine language
Graphics:	Bitmap graphics; everything that newer machines offer; graphics with whiteout and lighting effects
Colors:	White backgrounds; lighting effects
Quotes:	Transforms your Commodore into an Amiga look-a-like. —Final Cartridge III *ad, 1987*
	No doubt of many of you will have seen screen pictures of this game on a certain 16-bit machine, and thought, "what on earth would a 64 version turn out like?" The surprising answer to this is—extremely good indeed. —Defender of the Crown *review,* Zzap!64, *August 1987*

	GEOS transforms the new, sleekly designed Commodore 64C into a powerful, easy-to-use computer with icons, pull-down menus, dialogue boxes and bit-mapped graphics similar to those found on more expensive personal computers on the market today. —*Commodore press release, 1986*
	Convert *Lemmings* to the C64? Impossible. Or so it seemed. —*Interview with* Lemmings *developers, 1993*
Examples:	jesperjuul.net/c64/4/

Given that the C64 was released a mere two years after Commodore VIC-20, it was understandable for C64 owners to feel anxious: how long before their new computer would be replaced by an even newer model?

The 1985 dual launches of the Commodore 128 and Commodore Amiga presented two challenges for the C64. While the Commodore 128 did not make C64 programs run faster, it did have an eighty-column display. The Amiga system was the bigger contender, as it was originally envisioned as a high-end video game system and was by many accounts so far ahead of its time that no one, "not even Commodore," understood it.[1] It had a graphical operating system with multitasking, color graphics, digitized sound, and a hardware "blitter" for updating graphics quickly. Remember that in 1985, "serious" systems like the Macintosh were still monochrome or grayscale and could only run one program at a time, not multitask. The Amiga seemed like the C64 heir, especially when the consumer-oriented Amiga 500 came out in 1987. It turned out that Commodore really had tried to

discontinue the C64 in the fall of 1985, only to restart production.[2] Readers wrote worried letters to magazines, fearing their computer would be left behind.[3]

With this background, we can revisit the 1001 Crew demo from chapter 3 (figure 3.2) and see that it has a dual function: The demo demonstrates a novel technical trick, but by using the Tutankhamen image, iconic for the Amiga and the well-known Amiga program Deluxe Paint (discussed later), it also signals a desire to keep up with newer machines. In that way, it embodies two imaginaries, the C64 as a machine for exploiting hardware glitches and the C64 as a machine trying to keep up with the future. *Can the Commodore 64 keep up?*

Tired of entering commands to control your computer? An alternative to the textual interface had been emerging long before the advent of home computers. Douglas Engelbart's "mother of all demos" had demonstrated the graphical user interface already in 1968,[4] and Xerox had developed the expensive Alto (1973, figure 0.7) and Star (1981) computer lines, leading to Visi On for PCs (1983), Apple's Lisa (1983), Macintosh (1984), the Commodore Amiga (1985), the Atari ST (1985), Digital Research's GEM user interface for DOS (1985), as well as Windows 1.0 (1985). What to do when your C64 computer looked to be made obsolete by computers not only faster but with interfaces that seemed like the future? You would recreate the interface on the C64.[5]

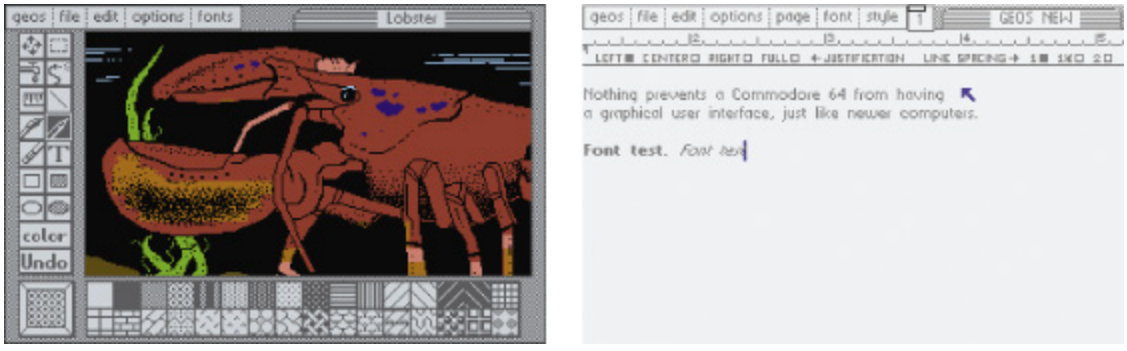


Figure 4.1 GEOS (Berkeley Softworks, 1986) gave the C64 a graphical interface to keep up with newer computers: Left: geoPaint. Right: geoWrite.

“Similar to More Expensive Personal Computers”: The GEOS Desktop

GEOS (figure 0.2d, figure 4.1), released in 1986, was a commercially successful attempt at upgrading the C64 to a graphical interface, sold both standalone and bundled with the cosmetically upgraded Commodore 64C. In the 1986 press release, Commodore emphasized exactly how the C64 was now keeping up as GEOS “transforms the new, sleekly designed Commodore 64C into a powerful, easy-to-use computer with icons, pull-down menus, dialogue boxes and bit-mapped graphics similar to those found on more expensive personal computers on the market today.”[6]

GEOS was surprisingly versatile, but the window and desktop system was limited. You could not actually move windows around like you might expect and GEOS required frequent switches between different floppy disks, but the system did facilitate mouse input, printer configuration, faster disk access, and graphical support for applications. The initial *64’er Magazin* review called GEOS “a programming masterpiece. The C64 is fully utilized with GEOS and is sometimes faster than you would expect from an 8-bit machine.”[7] The *Compute’s Gazette* review of GEOS emphasized both the attraction of graphical interfaces for a broad audience and how the C64 could now do what newer graphical computers could do.[8] The C64 was not designed for connecting a mouse, but by serendipity its connectors for “paddles,” the now almost-forgotten knob-turning input devices for games like *Pong* and *Breakout*, could be used for a mouse.[9] (► See GEOS on the website.)

Figure 4.1 shows the main GEOS programs, geoWrite and geoPaint. As an illustration of the C64 limitations, geoWrite provides multiple fonts and styles, but the screen is too narrow to fully show the eighty characters expected on a line. In the illustration, I have broken the line into two, such that the whole sentence can be seen. GEOS was likely subject to as much piracy as other programs, but we have an idea its popularity: a 1990 *Compute!’s Gazette* reader survey revealed that 56 percent of respondents used GEOS.[10] The survey also showed that the move from a textual interface (BASIC) to a graphical one did not erase the appetite for programming, as 80 percent of respondents reported to be programming, mostly in BASIC.

GEOS assuaged worries about keeping up—for a while. It was serviceable, but floppy disks still stored a mere 170 KB and GEOS quickly filled them with system files and applications. Still, the *64'er Magazin* editorial in January 1988 assured readers that they could just add extra RAM to their machine and that the C64 was still relevant, "The C64 is currently the most cost-effective entry option in computer technology, with applications that would challenge more powerful (and much more expensive) computers."[11]

Transform Your C64 into an Amiga Look-Alike: Final Cartridge III

If the GEOS desktop was not that useful by itself, the Final Cartridge III desktop was positively gratuitous. Final Cartridge III (1987) was a wildly popular cartridge that fixed central C64 flaws. There were numerous cartridges serving this niche, including Epyx Fast Load (1984) and the original Final Cartridge (1985). Final Cartridge III speeds up the disk drive, has its own faster tape format, has a reset button, and allows for seeing the contents of a floppy without overwriting the current program. Like other cartridges of the time, such as Action Replay (1985), it also features a "freeze" button for taking a snapshot of, and later restoring, the currently running program. The freeze button is ostensibly for backing up or saving your progress in a single-player game, but it is obviously useful for piracy too, even if it is so easy to use that crackers frowned upon distributing programs made using such a tool. For programmers, the Final Cartridge III features a "monitor" for disassembling and assembling machine language software and identifying graphics assets. It is a transformative cartridge that vastly improves the quality of life for a C64 user. Strangely though, Final Cartridge III starts your computer on a barely useful desktop, which consists of a few utilities—disk operations, preferences, calculator, and notes (figure 4.2), the styling of which is clearly based on the Amiga. Why was the desktop there? The manual openly explains the inspiration and introduces the reader to core concepts of windows, selecting, dragging, and so on: "The user interface, the windows and pull-down menus are based on ideas applied in the Apple Macintosh and the Commodore Amiga. . . . As

not everybody will know everything about these subjects, we first explain some general things."[12]

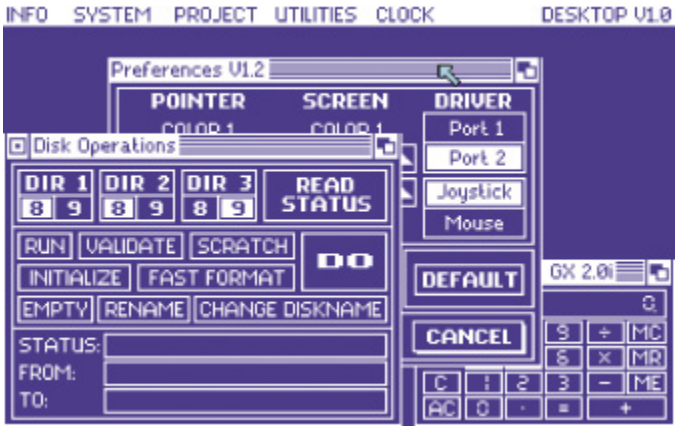


Figure 4.2 The Final Cartridge III (Riska B. V., 1987) combined fast tape and disk access, facilities for "backup," programming and graphics support, and an entirely gratuitous desktop functionality separate from the rest of the computer.

An ad promised that the cartridge "transforms your Commodore into an Amiga look-a-like."[13] After moving a few windows around, though, you would quit to the normal C64 BASIC interface to use the cartridge facilities while interacting with the computer like you were used to. The desktop was a gimmick, but cartridges like the Final Cartridge carried an imaginary for C64 users. With a Final Cartridge, the C64 was a machine for game players, programmers, and perhaps cheaters and crackers.

Keeping Up with the Games

Cinemaware's 1986 *Defender of the Crown* (figure 4.3) for the Amiga was not only billed as an "interactive movie" but contemporary reviews expressed unanimous awe about the quality of the graphics. Said *INFO* magazine, "This game's graphics have set new standards for computer games. The detail, color, and animation in this two-disk graphic adventure are breathtaking, and the music is impressive."[14]

What was so special about it? The "new standards" mentioned in the review referred in part to the abilities of the Amiga, able to display thirty-two colors at the same time (4096 in the special "HAM" mode) in a 320-by-

200 resolution, where the C64 could only show two colors per 8-by-8 cell. In 1986, the Macintosh was still black and white and the PC's graphics abilities were limited. Compared to other games of the time, like PC game *King's Quest III*, *Defender of the Crown* had higher resolution and more colors, but the frame of reference for the graphics was also different. Where *King's Quest III* used a cartoony style with monochrome surfaces and often black outlines, *Defender of the Crown* was more painterly, exploring light and shadow. In fact, the manual explicitly made claims about being inspired by more lofty things than video games:

> Welcome to the Master Designer Software production of *Defender of the Crown*. We are pleased to present this title as part of our new line of interactive movies which we call *CINEMAWARE*. We think you're going to enjoy it. Cinemaware derives its inspiration from the movies, not other computer games. The result: ideal entertainment for the mature player looking for greater challenges and a more "adult" experience.[15]

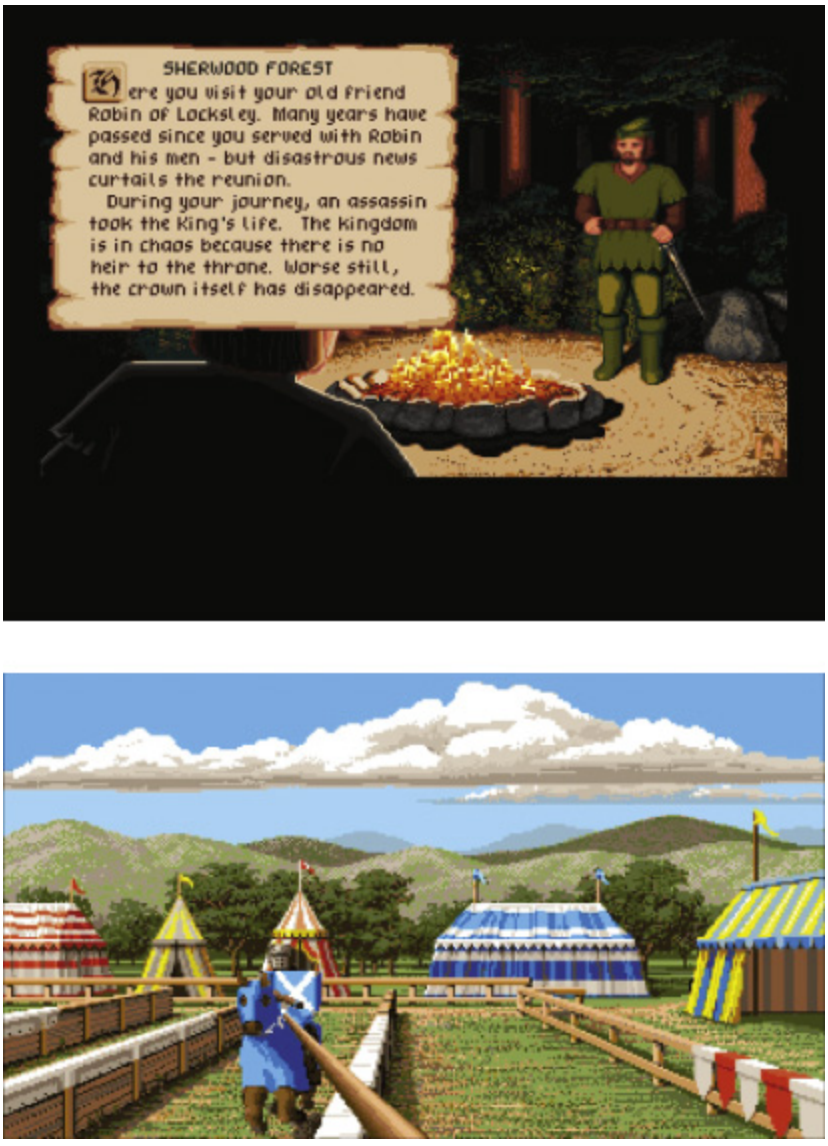


Figure 4.3 *Defender of the Crown* for the Amiga (Cinemaware, 1986), billed as an "interactive movie," "set new standards for computer games."

The 1987 C64 version (figure 4.4) naturally did not match the Amiga original, but it was well received: "No doubt many of you will have seen screen pictures of this game on a certain 16-bit machine, and thought 'What on earth would a C64 version turn out like?' The surprising answer to this is . . . extremely good indeed."[16] The Amiga's influence on the C64 was not technical, given that the C64 was more limited, but about composition and light: the meeting with Robin Hood in the forest features the central fire as the single light source, lighting up tree trunks, Robin, and the outline of the

characters. This leads to whiteout and black areas on the screen, which translate well to the C64.

Figure 4.4 *Defender of the Crown* for the C64 (Cinemaware, 1987). Could the C64 keep up with the Amiga? It was reviewed as “surprisingly” good.

Defender of the Crown was remade for numerous platforms. From 1985 to early 1987, the only platform that could match the static images of the Amiga was the Atari ST. Each C64 remake of a game from a newer platform would trigger the question posed in the review, “what on earth would a 64 version turn out like?” A successful remake could temporarily dampen fears of being left behind.

In 1991’s *Lemmings*, one of the most recognizable Amiga games, the titular lemmings—universally acknowledged as cute—walk blindly towards their deaths, unless the player can save them. Each lemming can be assigned one of eight roles, such as climbing, digging, or blocking other lemmings. Figure 4.5 shows the 1991 Amiga and 1994 C64 versions of the game. (*Lemmings* developer DMA Design later created *Grand Theft Auto*.)

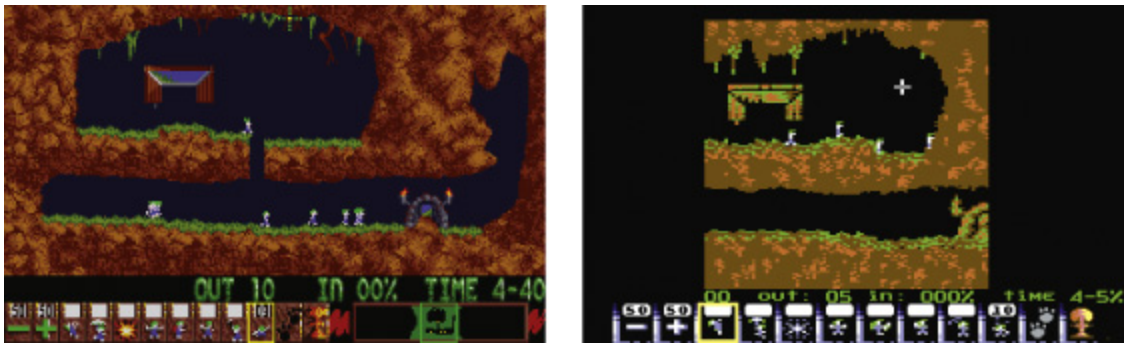


Figure 4.5 *Lemmings* for the Amiga (DMA Design, 1991; left) was one of the most famous Amiga games. Could the C64 conversion (E & E Software, 1994; right) overcome the C64's limitations? Unusually, the background is made from sprites, while the lemmings are made from characters.

Commodore Force magazine covered the development of the C64 version of *Lemmings* excitedly in 1993 with a three-installment developer diary. When the game was finally published, *Commodore Force* praised it, "C64 *Lemmings* is a spot-on conversion of a computing legend."[17] A flattering interview with the developers also noted the barriers to the conversion: "Convert *Lemmings* to the C64? Impossible. Or so it seemed. E & E software's conversion of the original Amiga classic is every bit as good as we'd hoped it to be."[18]

But how was the conversion made? The obvious barrier to a C64 *Lemmings* was the eight-sprite limit. *Lemmings* often has more than eight lemmings on a horizontal line, so sprite multiplexing could not solve the issue. However, having to draw the entire bitmap every frame (like the Amiga did) was outside the capabilities of the C64. Like in *Delta*, the developer reversed the role of sprites and characters. In *Lemmings*, the background is made from sprites, but the lemmings are made from characters that are updated in software (known as software sprites or "bobs"). These techniques, multiplexed sprites for updating a background and software bobs, were common in demos. (► See videos on the website.)

Demos: The Many Afterlives of Tutankhamen and the C64

The previous chapter opened with 1001 Crew's demo showing an image of Egyptian pharaoh Tutankhamen's death mask (1323 BCE), with the distinctive feature that the image was also shown in the screen borders, using the then-new sideborder technique. Why Tutankhamen, though? The image carried significance as Avril Harrison's iconic promotional image for the 1985 Amiga program Deluxe Paint, adorning the box of the software from version II. Tutankhamen became a recurring motif in C64 demos. The 1986 *Power Windows* demo gratuitously put the Tutankhamen image inside a simulation of the Amiga Deluxe Paint interface, letting users pull down the "window" (and nothing else). The 1989 *Road of Excess* demo (figure 3.11) combined the image with scrolling and with the new FLI technique (chapter 3) of showing more colors than the normal multicolor mode allowed.

The most direct demo reference to later systems was the 1986 C64 *Amiga Demo 2.0*, emulating the iconic Amiga *Boing* demo used in 1984 to demonstrate the system prerelease. Jimmy Maher has detailed the history of the Amiga demo and its iconic status, having been created in a rush before one of the first public demonstrations of the machine.[19] To this day, Amiga-related products often use the *Boing* ball as an Amiga signifier. To remake *Boing* and to include an Amiga logo on the C64 is a bit nonsensical, but again—it makes a claim that the C64 can keep up. The *Boing* demo was graphically and visually impressive for its time, but it also showed the Amiga's multitasking facilities by allowing you to drag down the screen and do something else while the ball would continue bouncing. The C64 remake goes back to the BASIC READY prompt once the ball starts bouncing, and by animating the ball using raster interrupts, it leaves time for regular BASIC commands and thus replicates, however primitively, the Amiga's multitasking facilities.

The demoscene was also changing. Many new things were easily possible on the Amiga, seen as the clear C64 successor in the demoscene, and Amiga demos like the 1993 *Desert Dream* included new 3D effects.[20] It was only around 1992 that the sheer processing power of newer CPUs allowed PCs to surpass the Amiga with three-dimensional games like *Wolfenstein 3D*, often credited with helping to define the first-person shooter genre. It seemed obvious that the C64 could not keep up, so demo groups reacted by demonstrating that, yes, it could, sort of. The 1995

Mathematica demo (figure 4.6) recreated the ball effect of the *Desert Dream* demo, a rotating 3D cube with fractals, as well as a simple version of the *Wolfenstein 3D* maze. Copying effects from newer machines became a common activity.[21] Given the C64's limited math abilities, it was wildly counterintuitive that these newer 3D and calculation-based effects would be possible on the C64. The more the reason to do them. Where early C64 demos were designed to update every frame (50 or 60 times per second), these later demos were accepted even with slowly updating effects. It was understood that the C64 could not really replicate what newer machines could do, but it could perform surprisingly well. (► See demos on the website.)

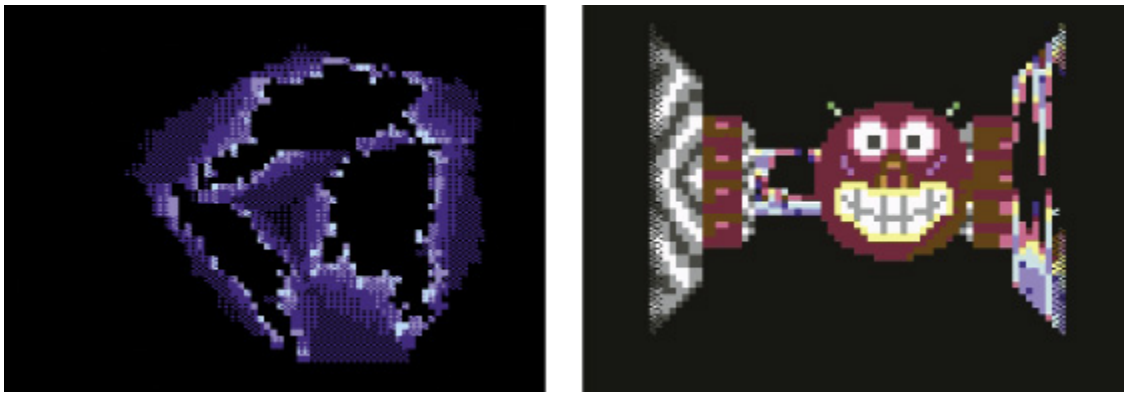


Figure 4.6 Later C64 demos were surprisingly good at recreating newer effects, if at lower resolution and speed. C64 demo *Mathematica* (Reflex, 1995) creates a Mandelbrot fractal image on a rotating cube (left) and a *Wolfenstein 3D*-like maze (right).

The End of Keeping Up

At some point, the time is up. Newer machines and games had inspired C64 developers to use the machine in a new way, applying the imaginaries of the Amiga to the C64 and considering how it would be possible to recreate a graphical interface or the action of *Lemmings*. The C64 posed very real limitations, but the flexibility of the VIC-II chip and the serendipity of the paddle controller interface allowed the C64 to keep up longer than expected. Still, a newer computer capable of showing eighty characters on a line is more attractive for word processing. A computer with a hard drive is better for desktops. A computer with improved, game-oriented graphical

abilities is easier to develop for. From the mid-1990s, there could be no more pretense that the C64 was as capable as newer computers. It was time to see the old machine’s limitations in a new light.

5

The Eternal Commodore 64 Style

Imaginary #5:	The limitations of the Commodore 64 are charming and should be emphasized rather than hidden. The Commodore 64 is not just physical—it can be remade and emulated. The Commodore 64 belongs to a different time, but it can play a role in the present.
Programming Language:	Machine language using newer machines as development platforms; C, other compiled languages; return to BASIC
Colors:	Everything
Graphics:	Character graphics; chunky multicolor sprites and monochrome sprites; PETSCII The default display device is now a modern monitor, clearly showing individual pixels.
Quotes:	The C64 is immortal. It will never die and it will never be too old. Never! *—mrsid, demoscene participant, 2011*
	I think we can all agree that original C64

	hardware is best, but it isn't without its problems. —Retro Recipes, *2019*
	The character set, provided by PETSCII and teletext, is used unadorned and rough cast, like concrete, which connects it to the Brutalist architecture. *—Raquel Meyers, 2017*
	Commodordion: A fully functional 8-bit accordion with bellows made out of floppy disks. *—Linus Åkesson, 2022*
Examples:	jesperjuul.net/c64/5/

Why Does This Look So Old?

If *Lemmings* was made to prove that the C64 could, sort of, keep up with the Amiga, the 2021 *PETSCII Lemmings* (figure 5.1) is a deliberately simplistic C64 port of the already anachronistic Commodore PET port of the 1991 Amiga game *Lemmings*. The C64 version does not even use the C64 color features. Why forsake all the technical knowledge and tricks developed over four decades and go back to this? The default PETSCII character set (figure 0.14) is now interesting *because* it fell out of favor in professional games around 1984. After the C64 as a universal computer, as a game machine, as a machine for exploiting unknown hardware glitches, and as a machine that keeps up with newer platforms, this latest imaginary rethinks the C64 as something entirely new: a historical machine whose

quirks and limitations are not to be hidden or worked around—they are now a cause for celebration.

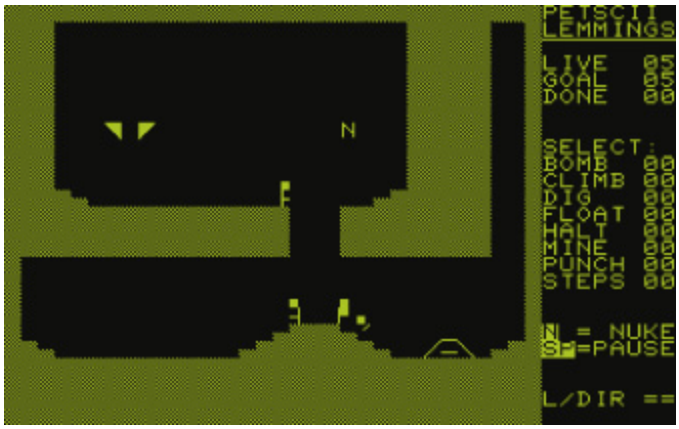


Figure 5.1 *PETSCII Lemmings* (Ko-Ko, 2021) recreates *Lemmings* using PETSCII characters and in monochrome. The C64's limitations and even the original character set can now be celebrated.

We see and hear new things that are extremely old: A PETSCII revival began in the 2010s, and there are now PETSCII graphics competitions, which, in detail, obviously cannot match C64 bitmap graphics, newer graphics modes, let alone modern computers, but which are interesting because they embrace even more extreme limitations than the C64 provides.[1] The $11 Music Compo in 2018 only allowed music employing the rarely used triangle waveform, usually seen as lacking the punch and flexibility of the square wave, giving an unusually soft expression to C64 music, with musicians "imposing artificial limitations on themselves for extra challenge."[2]

The final March 1994 editorial of *Commodore Force* lamented the death of the C64 as a commercial platform: "Times are hard for C64 gameheads. With a mere handful of commercial releases planned, even fewer chain-stores and independent computer stockists who intend to sell them, it's not hard to see why many 8-bit Commodore users are looking to other systems to provide the entertainment they crave."[3]

It took some years after that, perhaps ten, before the lack of commercial support ceased to feel like an issue and before the limitations of the C64 began to feel like a *style*—like something we can choose because we have other options at our disposal. Though style is sometimes understood to be

superficial, style plays a large role in the creation, circulation, and reception of art and media.[4] For example, games promoted as independent video games are characterized by an "Independent Style," where pixel style or hand-drawn graphics are used to signal that a game is different from big-budget game productions.[5] Using the C64 and emphasizing its limitations performs a similar function, signaling that you have deliberately chosen to use an old platform, and that you are a discerning nonmainstream computer user.

If PETSCII characters are simple, there is also an immediacy to them, as you can compose images by typing the moment a C64 is turned on. Spanish artist Raquel Meyers uses PETSCII characters as the subject and material for a series of artworks, performances, and workshops. Figure 5.2 shows a Meyers art performance using PETSCII graphics from a C64. For Meyers, using the Commodore 64, or other types of digital text like Teletext, connects to the history of typewriter-based concrete poetry, to the style of brutalism where the raw components of a building are visible, and to craft in general.[6] Meyers also converts PETSCII images to physical objects (figure 5.3) and cross-stitch, whose diagonal lines match the diagonal lines of common PETSCII characters. I interviewed Meyers about the meaning of using PETSCII. Meyers sees the use of old technology not as nostalgia but as a kind of resistance to technological change, reminding us that we do not need newer devices:

> **Jesper Juul:** Even if you were to use the best possible graphics program for the Commodore 64 right now to create graphics, it would be challenging due to the low resolution. But you've chosen to do something more extreme, with PETSCII which is even more constrained. Don't you have enough constraints in the Commodore 64? Why go that extra step?
>
> **Raquel Meyers:** Technology changes so fast that we don't have time to learn how to use it. The idea is that it doesn't matter how old a technology is. If it can teach you something or it can give you something, just use it. Workshop participants start to think about technology in a different way. You don't need a supercomputer in your super cell phone. You can use very simple devices.[7]

Figure 5.2 Raquel Meyers's art performance with C64 and PETSCII characters. March 3–5, 2017 // Square Sounds Melbourne 2017, Melbourne, Australia. (Photo by O-Zan Photography.)

Figure 5.3 Raquel Meyers: *PETSCII mosaic* (Keys of Fury / Le Shadok + FEFFS Strasbourg, 2016), translating PETSCII characters into physical objects, which are then made into mosaics. (Photo by Raquel Meyers.)

Meyers releases some of her artworks as demos as well under the handle AcidT*, unusually connecting an art practice to the demoscene. In the revival of PETSCII graphics, the C64's full-screen editor becomes interesting again for the immediacy it offers, and if PETSCII was originally a workaround for the lack of bitmap facilities in the Commodore PET, a crude way of simulating organic shapes using only a small number of characters, that crudeness is now charming.

In the demoscene, the 2020 ECM Mode Demoeffect Competition also embraced additional limitations and permitted only the use of the nearly forgotten *extended character mode* (ECM), which allows you to choose from only sixty-four characters (instead of 256), in return for allowing the background color of each character cell to be one of four colors.[8] The 2020 *Copper Booze* demo (figure 5.4) is both simple in the hardware facilities available to it and complex in featuring unusual rotating colored bands and effects—some derived from alternating colors per horizontal line. (► See demo on the website.) The demo is also unusual for being distributed with its source code. Where early demos makers were strongly protective of their tricks, many now approach demos as a culture of sharing. Going through the source code reveals a well-structured program, where the effects are not derived from hardware tricks but from algorithmic techniques, creating a custom character set that only approximates the slopes of the lines, yielding slightly irregular lines. The custom character set was made using an intriguing external program written in the Java programming language, suggesting a programmer whose day job is in a large corporation, used to making programs for other to read, and very distant from the sometimes underground pretenses of the early demoscene.

Figure 5.4 *Copper Booze* (Booze design, 2020) using the nearly forgotten ECM mode to create new effects, including rotating colored lines, embracing limitations but also creating something new, and sharing the source code.

Figure 5.5 Now that the C64 can be emulated virtually, the physical machine becomes interesting again. Linus Åkesson's *Commodordion* consists of two working C64s connected by a bellows made from floppy disks (Åkesson 2022).

Celebrating the Physical C64

Like in Meyers's performances, the attraction of the physical C64 is also present in Swedish musician-tinkerer Linus Åkesson's Commodordion project (figure 5.5), where Åkesson connects two C64s by a homemade bellows constructed from floppy disks to make a functional accordion. The sound is produced by the C64s with the volume controlled by measuring the airflow inside the bellows.[9] Like with the paddle port accommodating the

later mouse, there is serendipity: The chromatic, usually left-hand, bass side of an accordion consists of rows of keys placed diagonally, which corresponds to the placement of key rows on standard keyboards like the C64s. As the C64 is virtualized in emulators, the tactility of the physical C64 become correspondingly interesting. (► See video on the website.)

Virtualizing the C64

In his book *Back into the Storm*, engineer Bil Herd describes Commodore as a chaotic company that had failed to retain important engineers: “I found out that even though CBM had designed the most popular home computer, the Commodore 64, those engineers were now gone. We were like the owners of a work of art and yet had lost the recipe ourselves to some degree.”[10] In my interview with Al Charpentier, he expressed disappointment that he had delivered Commodore’s two biggest successes, the VIC-20 and the C64, yet Jack Tramiel did not allow him to make a third computer. Charpentier left.[11]

Commodore tried to create new but not upgraded versions of the C64, such as the portable SX-64, as well as a host of other machines. The only genuine attempt at upgrading the C64 was the Commodore 65, canceled in 1991 before release. The string of new Commodore machines—the ill-fated Plus/4, the 128, the Amiga, and Commodore’s PC line—were tied to the consistent belief that the C64 was about to be replaced, and it was therefore not upgraded, did not become faster, its graphics were not improved, and it did not gain more memory. Commodore produced only the 1986 REU RAM expansion, rarely used outside GEOS. Because it always looked like it was about to be replaced, the C64 remains a 1982 insect trapped in amber, forever unchanging.

What is a Commodore 64? It is clearly a physical device, or more precisely a group of connected devices. A fully equipped C64 is thirteen different physical objects:

1. C64 machine
2. Power supply
3. 1541 disk drive

4. Tape drive (the Datasette)
5. Utility cartridge (e.g., the Final Cartridge III)
6. Floppy disks
7. Tapes
8. Cartridges
9. Joysticks
10. Mouse
11. Television or monitor
12. Printer
13. Modem

The later life of the C64 has been about recreating every single one of these objects in hardware and software. The C64 itself has been virtualized through a range of emulators, from CCS64 to VICE, and the emulators support files representing the previously physical storage media like tapes, floppy disks, and cartridges. The line has not been blurred—we can easily tell the difference between a physical floppy and a file representing a floppy—the line is just unimportant in practice. We can connect a 1541 disk drive to a laptop, and we can convert physical floppy disks to files on our modern computer and vice versa. The Mouster lets us use USB game controllers or mice on a C64. Where early C64 culture was intensely physical, based on owning hardware, meeting physically, and shipping floppy disks by mail, it is now perfectly possible to be part of C64 culture without access to a machine.

The Chip of Theseus: When Is It Still a C64?

But what if we want to eschew software emulation and build a new C64 instead? Two popular home computers of the 1980s, the ZX Spectrum and the Apple II, were remade in numerous clones, the ZX Spectrum especially in eastern Europe, the Apple II in Asia and the Americas.[12] Similarly, IBM lost control of the IBM PC platform as competitors soon built IBM PC clones from off-the-shelf components. Even Commodore began building PC clones around 1984. Yet there were no C64 clones. Why not? This goes back to Commodore's role as a semiconductor company, a background that

it used to design chips for the C64 in-house, especially the VIC-II graphics chip and the SID sound chip—and those chips were not available from other suppliers. By comparison, the Apple II, Spectrum, and IBM PC were made mostly from off-the-shelf components, and a resourceful engineer could create clones.

The road to building a C64 without dedicated chips from Commodore was long—twenty-two or forty years. The first full C64 recreations were done with more modern ASIC and FPGA technology, chips that emulate the innards of the C64 chips in hardware. Jeri Ellsworth's 2004 C64 Direct-to-TV packaged a complete C64 recreation into the size of a 1980s joystick.

For the seemingly simpler route of recreating the physical machine, a whole community has been creating new circuit boards for placing in a C64 case. The 2023 EVO64 circuit board replaces the original board, but still requires Commodore chips or replacements to work.[13] The original chips have been out of production for thirty years but are now being recreated with new technology. The 2022 ARMSID can replace the SID chip on a circuit board, and the 2022 VIC-II Kawari replaces the VIC-II chip with additional features such as HDMI output, a modifiable palette, and switching between PAL and NTSC.[14] Randy Rossi, the Kawari developer himself, raises the question of whether a machine with this chip is still a C64, like the philosophical question about the Ship of Theseus, whose parts were gradually replaced, leading to the question of whether it is still the Ship of Theseus. For most, an original Commodore chip is the gold standard, and software emulation is somewhat inauthentic. Rossi argues that his VIC-II replacement is closer to the original chip because it electrically fulfills the role of an original VIC-II on the board.

A triumphant 2019 video on the Retro Recipes channel describes the challenge of building an all-new C64 with no parts made by Commodore. Channel creator Christian Simpson prefaces the challenge with a statement: "I think we can all agree that original C64 hardware is best, but it isn't without its problems."[15] Why is the original hardware "best," though? Though the video is about breaking free from Commodore's parts, it takes for granted an eternal authenticity imbued in the original Commodore chips.

Quality of Life

In all this, whether the C64 is virtualized, emulated, or emulated with custom hardware, there is a tension between the attraction of the original chips with their Commodore provenance, and a modern desire for "quality-of-life" upgrades. The original keyboard, while good for its time, is painful to use today with only two cursor keys and a high angle that will strain your wrists. If you can plug in a modern keyboard, you probably will. While the slowness of the C64 floppy drive was overcome with cartridges and modifications, you are more productive on a modern computer with a larger screen and modern comforts like versioning and backups. Even though a few third-party products such as SuperCPU can speed up a C64, developers almost exclusively develop for the original limitations of the machine, upgrading only their development tools. In correspondence, electrical engineering professor Stephen Edwards explained the joy of examining older hardware with new tools and skills: "I think it's stronger than just nostalgia. Not only do I enjoy doing things that I did when I was younger, but I'm much more skilled now and have access to much more equipment, including lots that simply didn't exist. So, it's not just reliving the past, it's doing it again far better than it ever was."[16]

In video game terms, this is like returning to an earlier part of the game with upgrades and better equipment. I interviewed Johan Grip, developer of the C128Neo circuit board about his design choices when making his Commodore 128 recreation (which includes a complete C64):[17]

> **Jesper Juul:** If you make enough changes, when would it cease to be a C128?
>
> **Johan Grip:** For me, the 128 is really the original board, per se. What I've done is no major surgery, I just added quality-of-life to it. So, if you want to do hardware modifications, you have a lot of functionality on the new board, which means you no longer have to add grabber connectors and little wires flying everywhere, etc.[18]

In a paper cowritten with Marleigh Norton, we noticed that video game design is characterized by two conflicting impulses: interface designers will (generally) try to make the interface of a game easy to use, such that basic tasks are easy to perform and feedback is consistent. However, game design

is also about making some tasks difficult to perform. Video game design is therefore a process of choosing what should be easy, and what should be hard for the player.[19] To use the C64 today, including recreating it in hardware and software, is to share this duality between the easy and the difficult. Programming the C64 is still challenging, but in modern development, we usually use another machine for developing the C64 program. The practical challenges of poor ergonomics and cumbersome file handling were central to much early development, but there has been a collective decision that such interface challenges are not attractive, though the C64s hardware limitations still are. Difficulty is an aesthetic choice, a choice about what we find interesting.

Nostalgia?

Is this fifth imaginary, dwelling on the C64 as a historical *style*, just *nostalgia*, then? Researchers Paolo Magaudda and Sergio Minniti argue that the concept of nostalgia misunderstands the reuse of old technology such as vinyl records and film photography. They argue that using old technology is not about reliving the past, but rather about assigning old technology a new authenticity and remaking it to a fit a new kind of use.[20] In their example, new vinyl records are usually heavier (180 grams) than original vinyl was, deepening the "tactile satisfaction" of vinyl, exaggerating the physicality of vinyl because that is what people find interesting—much like C64 software that emphasizes the machine's limitations. The old medium or technology you are now using is a reinvention of the original technology and a reinvention of how to use it, like with Åkesson's accordion or Meyers's use of the C64 in her art.[21]

This fifth imaginary, fifth life, does not erase the previous imaginaries. It is perfectly possible to continue to engage the C64 as a game machine, to create demos with new technical tricks, or to work hard to make the C64 keep up with newer computers. But the fifth imaginary provides a new framing for C64-related activities.

Similarly, it is not only that people use the C64 today because they are nostalgic for its heyday, or that they recreate the old experience of waiting twenty minutes for a game to load. Users are connecting the C64 to the internet, adding new virtualized devices, floppy disk replacements, and

emulators. Programs now load in seconds, and there are convenient 16 GB SD cards with the totality of all games and demos ever made for the machine. To use the C64 now, including when making C64 versions of newer game genres or programs, is to connect the C64 to the present and to make it something subtly different than it was. It is a way of changing the present and a way of imagining a different past, sometimes a past where games were better, where we were better at programming, or even where the roles assigned to us by advertising were less rigid.

Early cracker and demo culture worked with excessive self-promotion, declarations of status, connection, and rankings. Even in non-English speaking countries, English was the language of communication, projecting that you were not just mailing floppies to your teenage friends while living with parents—you were working on an international stage. Demoparties had similar English-language titles—The Party, The Gathering, Assembly. In the 2010s some parties switched to local languages, like the Swedish Gubbdata, started in 2012. The Gubbdata manifesto starts by declaring, "We are geezers with old computers."[22] "Gubb" means "old man" in Swedish, and "data" is a dated term for computing, thus signaling both old men and antiquated technology. The title does not change gender assumptions about who would be participating in such a party, but care has been taken to signal the local, the old, and the obsolete. A report on the 2022 Gubbdata party emphasizes the ailments of aging participants, "Atari guy has had an aortic aneurysm. That's an awesome word! I must remember that. Totally gubb!"[23] There is no returning to the early days of C64 demos, but the activity can transfer to present day along with the self-awareness of the current age of early participants.

The Missing 64 in History

This still leaves us with the question of why the C64 is missing from much computer and game history, even though it was the best-selling home computer of the 1980s, and even though it was the 1980s platform for which most games had been made. Part of the answer is that the US has forgotten is own computer and video game history, and part of the answer is that, while US-centric histories are best known, US and European C64

histories were very different, as the C64's commercial life lasted longer in Europe. We cannot say what caused what, but certain things coincide:

- The popularity of the Nintendo Entertainment System in the US (34 million sold), and its lower market penetration in Europe (8 million sold).
- The C64 was sold in more channels in Europe.
- Where European Commodore distributors sold C64 game packs, Commodore US refrained from openly promoting the computer as a game device.
- The demoscene in Europe was a technical culture that used the C64 for many years and generated news and excitement.

Commodore UK software operations head Gail Wellington argues that even Commodore's management failed to understand the situation, as "most did not recognise the significant differences in Commodore's reputation and market position between the US, where it was seen as a low-end opportunistic company, and Europe, where it was recognised as a major quality technology player."[24]

Video game and computer histories were later rewritten around a few figures whose stories were repeated. In the introduction, I showed how popular computer histories by Steven Levy (*Hackers*, 1984) and Robert X. Cringely (*Accidental Empires*, 1992), as well as the 1999 TV drama *Pirates of Silicon Valley* managed to erase most computer companies and people over the course of just fifteen years, leading to a simplistic story of Steve Jobs, Bill Gates, and only occasionally IBM. This is not just about Commodore, as even IBM's contribution has been forgotten amid the focus on Jobs and Gates—the PC has long been the most popular computer platform, but few people know who designed it.[25]

In video game history, once Nintendo had been established as a primary video game brand, it became easy to rewrite early video game history to ignore home computers. As a video game researcher, I have done this myself, using *Super Mario Bros.* as an example, even though I grew up playing *Boulder Dash*, *Monty Mole*, and *IK+*.

In his 1994 lament for the Commodore company, Thomas Halfhill describes the Commodore Amiga as the first “multimedia” computer, a popular term in the mid-1990s to describe a computer with both graphics and sound.[26] Commodore’s argument with the C64 and Amiga (and Atari’s with its 8-bit computers) was that a computer should have color and dedicated hardware for playing sound, moving objects, and scrolling the screen. These are now obvious required features of any computer or phone, but given how quickly computing developed, and given the size of the video game market, it seems incredible that there was a ten-year gap from the Atari 8-bit machines (1979) and the C64 (1982) before IBM PC-compatibles (or Macs) could do a barely smoothly scrolling platform game, the 1990 *Commander Keen in Invasion of the Vorticons*. What took so long? Just that IBM and Apple avoided updating their graphic architectures to do real-time graphics, afraid that color or fast-moving graphics would lessen the value of their machines in the public eye. IBM and Apple were afraid that their computers would be too much fun.

Vintage Computer Festival East

Wall Township, New Jersey, April 2022

It is a research trip, though a naive observer could see it as a pilgrimage, my journey to the Vintage Computer Festival East on the Jersey Shore. I am going where the fans go, to meet the now-older Commodore engineers who made the machine we revered. There are souvenirs and T-shirts. We are at the InfoAge Museum, in poorly maintained and disused military barracks, sundry radars and military equipment standing around between buildings, and a permanent exhibition of old computers going back to the 1950s. I am here for the second of this year’s themes: Computers for the Masses. Those in the know recognize the Jack Tramiel quote, “Make computers for the masses, not the classes,” and the festival features several panels on Commodore.

The festival combines talks, workshops, and an exhibition featuring personal collections of portable computers, of legendary computers like the SOL-20 and the Datapoint 2200. Private Commodore collectors show their C64s (extra excitement gleaned

for rare versions), VIC-20s, and earlier Commodore PET machines. The more obscure the better, visitors showing pangs of recognition for old machines and software (often games).

The emotional heart of the event is Saturday's panel of former Commodore employees reminiscing about their time at the company. There is something necessarily wistful, wishful, and tragic about Commodore, and about being a Commodore fan. For a brief period in the mid-1980s, it was the leading computer company in the world (by computers sold), and in the late 1980s it had the most audiovisually advanced computer, the Amiga. Then everything come to a halt. We waited for new computers and better advertising campaigns that always failed to materialize, and the company went bankrupt in 1994. There was (and is) an outdrawn aftermath of community infighting while new owners of the Amiga brand go bankrupt in turn.[27]

To be a Commodore fan now is to constantly second-guess infinite numbers of decisions made decades ago: what if US marketing had been better? What if Tramiel had stayed? What if any logical improvement of the C64 had come out on a reasonable schedule—1984, 1985, and the Commodore 65 hadn't been canceled? What if the Amiga engineers had received sufficient resources (and direction) to develop the new chipset in time? There are legendary prototypes and chipsets for which fans hunger for information.

The twelve-person panel takes turns telling Commodore stories and asking each other about historical events. There are moments of pause at the mention of those not here—most for the charismatic and departed—Jack Tramiel and Chuck Peddle.

The biggest laughs are reserved for tales of daring engineering—the last-minute kludge of components in unusual configurations, like the phase-locked loop that stabilized the C64 video signal. Chuckles and laughs abound among the gathered engineers. There are tales of unconventional, nonconformist, and brilliant engineering—the kind that, all agree, would not have been accepted at any other company.

There is planned brilliance, like in Al Charpentier's tale of proposing the VIC-II chip to Jack Tramiel who immediately approved, in the tale of the physical layout of all the transistors and components, and in how the chip had only one flaw in its first iteration, and only needed one more take to be done.

The foe is almost any part of management from the departure of Jack Tramiel in 1984 to the company's demise. There are stories of dedication, of engineers sleeping in their offices for weeks at a time, one cocooned in bubble wrap.

Commodore's status is thus both as a technological powerhouse, as a mass-market company, and as a producer of technology where it is possible to name the progenitor of individual lines of programming or transistor layouts. The company now represents a longing for a simpler time, where one person or a few friends really could design a new chip or computer. That longing for a simpler time runs parallel to ideas of what the present might have looked like, had Commodore survived. The company would be making computers as complex as the ones that exist today, made by teams of hundreds or thousands of engineers, but—and we do not need to think this through—modern technology might yet have had that Commodore *je ne sais quoi* of daring and personal engineering.

Audience questions show that the dedicated have read the 1985 *IEEE* journal article on the C64's design, Bil Herd's 2021 autobiography, and Brian Bagnall's three-volume history of Commodore, in addition to attending earlier events and watching available videos. (Writing this book has made me one of those people.) For some topics, questions are not entirely resolved. Engineers have had successful careers since, and the original impetus behind a 1981 decision may have been obscured by later events. It is the fans for whom all the technical features or historical events are most present.

Commodore 128 engineer Bil Herd leads the panel, loud and fun, and often making fun of the legendary brashness of his younger Commodore self. At the end, Amiga engineer Dave Haynie, loved in the community for sharing his story and design

> documents for projects realized and unrealized, is close to tears over the mismanagement of the company, and the greatness/community/vision lost, and the audience shares the moment with him.
>
> There is a wistfulness, a happy melancholy flickering into an alternative world. There are long applauses and a rush to get the final group photo. There is pure fandom, as Commodore employees are asked to sign books, computers, and motherboards. There are those who hang around, hoping to speak to their heroes.

What the C64 Did

Should users be able to program? Joy Lisi Rankin laments that we are no longer computing "citizens," like early BASIC machines were teaching us to be.[28] We consume computing, but we no longer learn to program. Michael Black has argued that the ideal of the "transparent" interface, like the graphical interface, had led to machines becoming opaque. In Black's telling, there were two strands of promoting new computers in the 1980s, one the IBM PC promising (and requiring) computer literacy, the other promising that "the Macintosh's usability model effectively made computer literacy obsolete."[29] As we've seen, GEOS coverage similarly promised that the new interface would make the C64 accessible to a broader audience uninterested in learning arcane commands.

Given the success of graphical user interfaces, going into this project, I was expecting to find a historical trajectory of moving away from programming on the C64. The trajectory I found was quite different. C64 magazines, even those dedicated to games, increased their coverage of programming topics. The documentation of C64 culture shows a late shift away from commercial software and towards programming, both in reaction to the growth of the demoscene, in the practice of POKEs and cheats, and in reaction to the slow death of the C64 as commercial platform.

The demoscene took computing more seriously, and was bent on building a community, and though the demoscene can be seen as subversive, it was (and is) focused on building institutions—recurring events, magazines, documentation, and databases—that tie together participants. Some of this was done using municipal support and other systems aimed at citizenship.

In this way, the demoscene fulfilled the promise of BASIC, and some of the lack that Rankin addresses. The demoscene created a group of people who understood computing, and it embodied a counterintuitive equivalence between underground computing and citizenship. The demoscene continues to find new C64 tricks to this day. *We could have wished* that a larger and more diverse group of people would be programming for the fun of it and sharing it. Unfortunately, Commodore advertised programming as a boys' activity, and there was wide cultural agreement that programmers were weird and nerdy boys.

The C64 did not change, it was not upgraded, but during its lifetime it appeared as an entirely different machine with entirely different imaginaries, entirely different answers to the question, "What is this thing for?" Each new imaginary gave the C64 an extra life. It was a universal computer for utility and computational literacy using the BASIC language; a machine for arcade and later games; a machine for subverting expectations and intentions and becoming part of a technical culture; a machine rushing to keep up with newer technology; and finally, a machine with a style to be chosen at will, with now-charming large pixels, analog synth sound, and other now-fascinating limitations.

I developed this five-step structure of imaginaries to express how the C64 changed, but I think it describes a more general pattern of how technology is approached, though every technological history will have different emphases:

1. *Looking back*: A new technology (or medium) is designed and is first used as a backward-looking continuation of previous technologies and practices.
2. *As promised*: Users get to know the technology and use it (among other things) for the kinds of new applications it was designed for.
3. *For its own sake*: Users study the technology for its own sake and find in it entirely new possibilities, against the intentions of the designers.
4. *Keeping up with the future*: The technology appears dated, and attempts are made to keep it on par with newer technologies and imaginaries.
5. *Historical style*: The technology is sufficiently old that it feels like an interesting historical style with no need to keep up.

Which is more important, the imaginary that we use for understanding the machine or the resistance that the machine puts up to what we are trying to use it for? We can easily default to rooting for one or other side of the equation, feeling that we must pick between *social determinism* (imaginaries; technology is defined by how interpret it) and *technological determinism* (resistance; technology determines everything), but I believe that not picking sides made it easier to understand the history of the changing C64.[30]

How did this happen? There was no inevitability in the history of the C64, and its history does not provide a recipe for how to make a successful computer, but we can now see how it concretely transpired: it took time for users to understand what the C64 could do. The machine was designed to be flexible. The SID sound chip allows developers to change waveforms on the fly, and its characteristic arpeggio-based music came in part because cost-cutting reduced the number of voices from four to three. The VIC-II graphics chip can create graphics in many ways. I counted fifteen methods for drawing a horizontal line, and this flexibility meant that what was considered possible was constantly changing. Furthermore, the machine was designed at a time when it was up to the user and programmer not to crash it, and there was no reason to prevent them from using it in ways outside the specification. If you manipulated the scroll-related registers in unsanctioned ways, entirely new effects appeared, and some of these were instrumental in creating the demoscene. The music of the SID chip provided the soundtrack, it allowed demos to come across as relevant experiences, commenting on and framing the often-unpolished graphics with synthesizer music. Music also assigned games moods and identities that the sometimes too-difficult gameplay could not convey on its own. The C64 lived so long in part because it was so successful from the outset, meaning that its obvious flaws—slow tape, slow disk drive, simplistic BASIC—created market opportunities and were remedied by add-ons, especially cartridges. The flaws were there due to the extreme speed at which the C64 was developed and because of extreme cost-cutting leading to the limited BASIC. When the flaws were fixed by third parties, this also gave the expectation that the C64 could continue to grow and improve.

The C64 also lived so long and became so many different things because Commodore, unlike most other manufacturers, was not afraid of color, sound, fast-moving graphics—games. There were many serendipities, and

the small decisions of national distributors affected whether the C64 was popular (Italy) or barely sold (Japan, France, Spain). It lived so long because, as we saw with the game charts, it was the only home computer to be successful in both the US and Europe. Finally, it lived so long because it was *cheap*—much cheaper than IBM PCs or Apple IIs, though not as cheap as ZX Spectrums. Many people could afford it, and the machine was therefore approached with many different ideas.

The C64 has not exactly been lost to history, but there have been too many histories that miraculously manage to ignore the most popular computer of the 1980s. I have made the case for computer histories that do not just focus on the US home market and that remember the role of the C64 and other computers. We can also write better video game histories, ones that do not overly focus on console games and histories that remember that the majority of 1980s games were on the C64 platform.

We used to talk of home computers entailing three things: 1) there would be computers in the home in addition to in the office, 2) home computers would be less powerful and expensive than serious office computers, and 3) home computers were fun, with colors, games, and fast-moving graphics, and "serious" computers could have none of those things. All computers are now like Commodore 64s and Amigas, featuring colorful, fast-moving graphics and sound, allowing us to play games. I will not claim that the C64 is the "most important" computer of its time, but the C64 defined an important part of what a modern computer is, and yet was ignored in computer history exactly because of its contribution of fast-moving color graphics, sound, *fun*.

The C64's continued demonstrations of what a 1 MHz computer platform can achieve does question our obsession with upgrades of computers and phones. There are historical and organizational reasons why software production happens the way it does, and millions of people are currently working to optimize software and hardware such that programs run faster and consume less energy, but the C64 shows that machines can have a much longer life than they do now. If we are to respect that extended life, perhaps we should take inspiration from the subversive mood of the demoscene and see the continued use of an old platform as what it is: a resistance to the wasteful planned obsolescence of the tech industry with its endless pressure

to upgrade, an appreciation for an old device, and the ingenuity to always find new things do with it.[31]

The Commodore 64 taught us that a computer can be fun and have many lives.

Glossary

6502: Family of CPUs—the main logic chip of a computer—designed by a MOS Technology team led by Chuck Peddle, released in 1975. Designed as a lower-cost competitor to Motorola's 6800 chips. Commodore bought MOS Technology in 1976. The 6502 family was used in Commodore's KIM-1 board, PET, VIC-20, and C64, as well as the Apple II, Atari's 8-bit series, the BBC Micro, and the Nintendo Entertainment System.

6510: The C64's CPU, a variant of the 6502 family featuring extra input/output used for controlling the tape drive and the memory map of the C64.

Amiga: See **Commodore Amiga**.

Amstrad CPC: 1984 home computer, competitor of the C64 and ZX Spectrum, popular in some parts of Europe. Sold as Schneider in Germany.

Apple II: 1977 home computer for connecting to a TV. The Apple II could be opened and tinkered with, catering in part to a hobbyist engineer market, but was later sold as a business computer. It was most popular in North America.

Arcade game: Arcade games were originally standalone and coin-operated games, but arcade games also came to mean a genre of games with fast and easy-to-grasp gameplay, usually centered on whether objects collide. The C64 was designed in part to play arcade games.

Assembly language: The human-readable form of **machine language**.

Atari 2600: 1977 game console.

Atari 8-bit: 1979 home computer series, ahead of its time with facilities for fast-moving graphics and sound.

Atari ST: 1985 16-bit machine featuring color graphics, sound, and a MIDI connector for music.

BASIC: The C64's default programming and interface language. BASIC was first developed at Dartmouth college around 1960. The C64 features a version of Microsoft BASIC, Microsoft's first commercial product.

it: A single unit of memory that can be set to 0 or 1.

itmap: Graphics where the color of each pixel can be set individually.

order: The C64's display features a central area of 40-by-25 characters surrounded by a border, light blue by default. The top/bottom trick disables the upper and lower border, while the sideborder trick disables the left and right border.

yte: A group of 8 bits. This corresponds roughly to a letter or number.

haracters: The 8-by-8 pixel units of the textual Commodore 64 screen, often containing literal characters such as A, sometimes PETSCII, and sometimes user-defined characters for creating the landscape of video games. On some other systems, these units were known as tiles.

harpentier, Al: Designer of the VIC-II graphics chip of the C64.

hip or **integrated circuit**: First developed around 1959, integrated circuits enabled modern computing by packing large numbers of electronic components (such as transistors and resistors) into a small package. The three most characteristic chips of the C64 are the 6510 CPU, the VIC-II graphics chip, and the SID sound chip.

IA: 6526 Complex Interface Adapter chip controlling timers, keyboard, disk drive, and other input and output of the C64.

ircuit board (also **PCB**): The board onto which electronic components such as chips are placed and connected. Opening a C64 reveals the circuit board with components.

olors: The C64 features a fixed palette of 16 colors. These are: black, white, red, cyan, purple, green, blue, yellow, orange, brown, light red, gray 1, gray 2, light green, light blue, and gray 3.

ommodore 64 (**C64**): Commodore's 1982 home computer. More popular in Europe than elsewhere.

ommodore 128: Commodore's 1985 home computer. Combined three computers in one, one of them a C64, one a special 80-column mode, and one with the CP/M operating system.

ommodore Amiga: Commodore's 1985 follow-up to the C64, considered ahead of its time for featuring a 16-bit CPU, higher resolution, hardware support for quickly drawing graphics, and a multitasking operating system

with windows and a mouse. First became successful when packaged as the consumer-oriented Amiga 500.

PU: Central Processing Unit—the chip executing a program by doing logic and calculation. The C64 CPU is the 6510, a variant of the 6502.

rack intro (also **cracktro**): Small audiovisual program run when a piece of software is launched to claim credit for cracking the software. Evolved into demos.

racking: Removing the copy protection from a piece of software, such that it becomes easy to copy and distribute.

atasette: The name for Commodore's C64 tape recorder for storing programs and data.

emo: Audiovisual program demonstrating technical and design skills, including finding new hardware glitches such as sideborder sprites.

emoparty: Meeting in the demoscene, often with competitions for the best demo, graphics, and sound.

emoscene: Originally mostly German and northern European culture of creating demos, often meeting at demoparties.

isk drive: Compared to the cheaper Datasette, disk drives used floppy disks and were the more expensive option for storing data and programs but were essential for programming and productivity. As far as we know, disk drives were popular in the US, Germany, and northern Europe but less so in the UK and Italy.

iskmag: Disk magazine distributed on floppies, commenting on cracking and the demoscene, often with charts naming the best demo and cracker groups.

inkel, Andy: Commodore 64 software product manager, developer, and coauthor of the *Commodore 64 Programmer's Reference Guide*.

loppy disk: The flat rectangular disks used for storing programs and data on a disk drive.

rame: Shorthand for the image the computer shows (50 or 60 times per second on a TV).

ame machine: Dismissive term for a computer, typically a home computer, used for playing games. In the 1980s, some manufacturers actively avoided

adding game features (such as color, sprites, and scrolling) to avoid the stigma of game machines.

roup: In cracking and the demoscene, members usually chose pseudonymous “handles” and worked together in groups.

erd, Bil: Principal designer of the Commodore 128 computer.

ome computer: From the late 1970s to early 1990s, there were computers designed specifically for use in the home, as opposed to for the office. These were sometimes dismissed as game machines. See also: **Personal computer**.

z: Times per second. A 50 Hz display draws the entire screen 50 times a second.

BM PC: IBM’s 1981 home computer, black and white and with no game-related features, unlike the short-lived home version IBM PCjr (1984) featuring color and **raster** features.

naginary: Any imagined concept of a technology’s function, context, or meaning. We design and approach technology using imaginaries.

B or **kilobyte**: 1,024 bytes.

lachine language or **machine code**: The instructions that the computer’s CPU understands directly. When using development tools, the term **assembly language** is used to emphasize the human-readable form of the language.

IB or **megabyte**: Roughly 1,000,000 bytes.

lemory map: The organization of a computer’s memory. At startup, much of the C64’s 64 KB of RAM is invisible to the CPU and most be made visible by hiding the Kernal ROM, BASIC ROM, or the registers of the chips.

lensch, Bill: Chip designer, part of the 6502 team, and designer of the later 65816 chip.

lHz: 1 million times per second. The C64’s CPU runs at 1 MHz.

licrocomputer (or **micro**): Common UK term for a **home computer**, or any computer that is not an office computer or mainframe.

lSX: 1983 computer standard developed by ASCII Corporation. Popular in Japan, and in some European countries such as Spain.

lultiplexing: See **sprite multiplexing**.

intendo Entertainment System (**NES**): Nintendo's 8-bit home console, released in Japan in 1983, North America in 1985, and Europe and Oceania in 1986/1987. Much more popular in North America than in Europe.

TSC: North American analog television standard, showing images 60 times a second.

AL: European analog television standard, showing images 50 times a second.

eddle, Chuck: Designer of the 6502 family of CPUs and engineer at Commodore. Convinced Commodore to create the Commodore PET. Wanted to design business computers but was overruled by Jack Tramiel in favor of the VIC-20, leading to Peddle's departure.

EEK: BASIC command for reading the content of a byte in the C64's memory.

ersonal computer: Computer used by one person, as opposed to a multiuser mainframe accessed via teletype or terminal. **Home computers** and **personal computers** overlap, but "personal computer" is typically used to signal being more "serious" than "home computer."

ET: Commodore's 1977 computer, designed as an appliance with integrated monitor and tape drive.

ETSCII: The graphical characters, originally designed for the PET, featuring card suits, lines, blocks, and other elements for composing a screen. Popular in the beginning of the C64's lifetime, later disused in favor of newer kinds of graphics, but popular again since the early 2010s.

ixels: The dots that constitute an electronic display.

OKE: BASIC command for changing the content of a byte in the C64's memory. Used for graphics and sound as well as for cheating in video games.

AM: Random Access Memory—the memory used to remember things, such as the pixels on a screen or the running program. The C64 has 64 KB of RAM = 65,536 bytes. This is equivalent to 32 pages of text. This was more than competing computers when the C64 was launched.

aster: The raster beam of a television draws the image from left to right, line by line, and moving down from the top of the screen.

aster interrupt: Allows you to specify a piece of program that runs when the raster beam has reached a specific line on the screen. In the 1970s and 1980s, only game-oriented computers featured raster interrupts.

OM: Read-only memory—the fixed memory in a computer, used for storing the operating system in home computers, typically including the BASIC programming language.

crolling: The feature allowing the screen to "pan" around a larger world. Supported by the hardware of the C64 and Atari 8-bit computers, and later the Amiga.

ID: The sound chip of the Commodore 64, similar to analog synthesizers of the late 1970s, featuring three voices and waveforms that could be switched continuously allowing for expressive music. Bob Yannes designed the chip with the intention of creating synthesizers.

ideborder: See **border**.

inclair, Clive: UK inventor and entrepreneur behind Sinclair Research, developed the ZX 81 and ZX Spectrum.

prite: To enable objects quickly moving around a screen, game-oriented systems of the late 1970s and 1980s allowed developers to specify the position of small graphical objects independently of the graphical background. Of early home computers, the C64, TI-99 4/A, MSX, and Atari 8-bit computers featured sprites. The VIC-II chip lets programmers specify the position of eight sprites, but more can be shown using sprite multiplexing.

prite multiplexing: By moving sprites while the screen is being drawn, more than eight sprites can be shown (though never more than eight on a line).

erminal: From the 1960s, large computers were often accessed by many users at the same time using terminals, first teletypes which printed output on paper and later terminals with screens.

I-99/4A: 1981 Texas Instruments home computer.

ramiel, Jack: Auschwitz survivor and president of Commodore until departing in 1984. Famous for slogans like, "Computers for the masses, not for the classes" and "Business is war." After Commodore, he took over Atari.

RS-80: RadioShack's 1977 home computer for connecting to a TV.

IC-20: Commodore's 1980 home computer, predecessor to the Commodore 64 with less memory and limited graphics and sound capabilities. Promoted as "The Friendly Computer" with ads featuring William Shatner, this was the first computer to sell a million units.

IC-II: The graphics chip of the Commodore 64, featuring characters, bitmaps, and sprites, designed by Al Charpentier. Originally designed for use in video game machines.

annes, Robert "Bob": Designer of the SID sound chip.

X Spectrum: UK computer launched in 1982, lower-cost competitor to the Commodore 64, hugely popular in the UK and in some parts of Europe. Repackaged as Timex Sinclair 2068 in North America.

Notes

Introduction

1. Wierzbicki, “Longevity of Commodore 64, VIC 20 Questioned.”
2. Jaroslav Švelch has similarly discussed the role of ZX Spectrum fans (“Keeping the Spectrum Alive”).
3. Reunanen, “Studying Yourself—Case Demoscene.”
4. “Best-Selling Desktop Computer.” Guinness assigns this title because the C64 was never upgraded and therefore is “the most popular single model of desktop computer”; Steil, “How Many Commodore 64 Computers Were Really Sold?”
5. “Home computer” here refers to platforms targeted at home use, unlike today where common platforms are found both in the home and in professional settings. For this reason, I am not including sales of DOS or PC machines, also given that we have no independent sales figures for the home. For MSX and C64, recent sources have downgraded earlier sales figures, but for the other platforms, I am using current Wikipedia figures as a proxy for the consensus. There are also conflicting figures for MSX sales, from 4 to 9 million. I refer to recent Japanese sources that put the figure at 4 million. Nishi, “[Legendary PC MSX] The Creator Finally Reveals the ‘Essence of Failure.’”
6. Steil, “How Many Commodore 64 Computers Were Really Sold?”
7. Kent, *The Ultimate History of Video Games*. I use “video games” to include all digital games played on consoles, computers, phones, and other devices.
8. Donovan, *Replay*.
9. Navarro-Remesal and Pérez-Latorre, *Perspectives on the European Videogame*, 15.
10. Haigh, “Becoming Universal.”
11. Haigh, “The History of Information Technology,” 452.
12. Halfhill, “RIP Commodore.”
13. Ceruzzi, *A History of Modern Computing*.
14. Kuorikoski, *Commodore 64*.
15. Image credits for illustration (via Wikimedia): Bilby: Apple II (License: Creative Commons Attribution 3.0 Unported). Bill Bertram: ZX Spectrum (Creative Commons Attribution-Share Alike 2.5 Generic). Binarysequence: TRS-80 (Creative Commons Attribution-Share Alike 4.0 International). Chi Ho Chan: Macintosh (Creative Commons Attribution 2.0 Generic). Chris Rand: Pong (Creative Commons Attribution-Share Alike 3.0 Unported). Doppelgangland: MSX (public domain). Ecksemmess: Datapoint 2200 (Creative Commons Attribution-Share Alike 3.0 Unported). Evan Amos: Atari 2600, Atari 800, Commodore 64, Commodore Vic-20, NES (public domain). Felix Winkelnkemper: Atari ST (Creative Commons Attribution-Share Alike 4.0 International). Journey234: ZX 81 (public domain). Michael Hicks: Xerox Alto (Creative Commons Attribution 2.0 Generic). Michael Holley: SOL-20 (public domain). Michael Holley: TV typewriter (public domain). Museoscienza.org: Amiga (Creative Commons Attribution-Share Alike 4.0 International). OliverGalvin: Commodore PET (Creative Commons Attribution-Share

Alike 2.0 France). Raimond Spekking: Commodore Calculator (Creative Commons Attribution-Share Alike 4.0 International). Rama & Musée Bolo: IBM PC, KIM 1, TI-99 4/A (Creative Commons Attribution-Share Alike 2.0 France). Rama: Micral (Creative Commons Attribution-Share Alike 2.0 France). Sammlung der Medien und Wissenschaft: Olympia typewriter (Creative Commons Attribution 4.0 International). Stuart Brady: BBC Micro (public domain). Wolfgang Stief: Computer terminal (public domain). Museo della Scienza e della Tecnologia "Leonardo da Vinci": Programma 101 (Creative Commons Attribution-Share Alike 4.0). FrakogeneX: PC-8001 (Creative Commons CC0 1.0 Universal Public Domain Dedication). Bill Bertram: Amstrad CPC 464 (Creative Commons Attribution-Share Alike 2.5 Generic license).

16. Gazzard, "Remembering Repton," 109; Kim Justice, "Commodore 64 vs ZX Spectrum."
17. Halfhill, "The MS-DOS Invasion: IBM Compatibles Are Coming Home."
18. Morningstar and Farmer, "The Lessons of Lucasfilm's Habitat," 19.
19. De Bartolo and North, "The Computer Company President of the Year."
20. Levy, *Hackers*. The erasure of Commodore in computer history is also discussed by Bagnall, who is critical of Cringely, *Triumph of the Nerds*, and *Pirates of Silicon Valley* (*On the Edge*, introduction.)
21. Sen, *The Triumph of the Nerds*; Cringely, *Accidental Empires*, 184.
22. A pillar of the simplistic standard story is the assertion that among the 1977 "trinity" of the TRS-80, the Commodore PET, and the Apple II, the latter must in some way be wholly unique. For example, Isaacson puzzlingly claims it was "the first personal computer to be simple and fully integrated" (*The Innovators*, 353). *The Triumph of the Nerds* equally strangely describes Apple as "the first mass market PC Company."
23. Burke, *Pirates of Silicon Valley*.
24. Isaacson, *The Innovators*.
25. The US- and Apple-centric bias of many histories is also noted by Sumner, "What Makes a PC?"
26. Wozniak and Smith, *iWoz*, 166.
27. It is possible to argue that because the Datapoint 2200 was not based on a microprocessor, it was not a "microcomputer," but I find this to be of minor importance. Its instruction set formed the basis for Intel's x86 architecture, and from an end-user's point of view, it functioned just like any other preassembled computer of the 1970s.
28. Rankin, *A People's History of Computing in the United States*, 1.
29. Andrews, *TOO BLUE!*
30. Bagnall, *On the Edge*, 58.
31. Peddle claims that Robert Heinlein inspired him to build a computer, but Steve Wozniak claims that Peddle copied the idea from Apple (Wozniak, 195). Peddle in turn claims that Wozniak "did not have anything to do with creating the personal computer" because he did understand the value of BASIC (Peddle, "The 6502 to the PC"). Bill Mensch reports having proposed a home computer to Chuck Peddle already in 1973 (Bill Mensch, interview by Jesper Juul, May 6, 2022; Mensch, "In Memoriam"). Realistically, the idea was in the air in the mid-1970s.
32. Tramiel and McMurtrie, "Early Commodore History."
33. Robert Heinlein's 1957 novel *The Door into Summer* is not focused on computers, but on cryosleep, robots, and time travel. It does contain an electronic drafting table. Stories about the one spark that led to an invention are generally simplifications.
34. Kubrick, *2001: A Space Odyssey*.
35. "Most Important Companies."
36. Haigh and Ceruzzi, *A New History of Modern Computing*, 181.

37. Tomczyk, *The Home Computer Wars*, 44–45.
38. Tomczyk, 195.
39. Arnold, “Twenty-Two Columns of Lowbrow Revolution.”
40. Tomczyk, *The Home Computer Wars*; Pleasance, *Commodore The Inside Story*; Herd and Morabito, *Back into the Storm*; Tyschtschenko and Kloeter, *Meine Erinnerungen an Commodore und Amiga*.
41. Bagnall, *On the Edge*; Bagnall, *Commodore: The Amiga Years;* Bagnall, *Commodore: The Final Years*.
42. Tomczyk, *The Home Computer Wars*, 162.
43. Bagnall, *On the Edge*, 468.
44. Tomczyk, *The Home Computer Wars*, 63, 300.
45. Wozniak and Smith, *iWoz*, 195.
46. Tramiel, “An Inexcusable Failure.”
47. Bagnall, *On the Edge*, 279.
48. Bagnall, 355.
49. Clarke, *The Collected Stories of Arthur C. Clarke*.
50. Kirby, “The Future Is Now,” 45.
51. Parikka, *What Is Media Archaeology*, 45.
52. Kluitenberg, “On the Archaeology of Imaginary Media.”
53. Kirby, “The Future Is Now,” 47.
54. Kluitenberg, “On the Archaeology of Imaginary Media,” 48.
55. Natale and Balbi, “Media and the Imaginary in History,” 203.
56. Bennett, *Vibrant Matter*, viii, 5.
57. Albert Charpentier, interview by Jesper Juul, April 22, 2022.
58. Perry and Wallich, “Design Case History,” 48.
59. Perry and Wallich, 49.
60. Charpentier and Herd, “40th Anniversary of the Commodore 64.”
61. Charpentier and Herd.
62. Bill Mensch, interview by Jesper Juul, May 6, 2022.
63. Wozniak and Smith, *iWoz*, 161.
64. The 6502 can only perform operations on 8-bit values, and the stack-pointer is 8 bits rather than the 16 bits of the 6800 chip, meaning that programs can only be quite simple. 6502 designers imagined a difference between the simple control systems they were designing a chip for and more complex computers which would need better math abilities and the ability to run more complex programs. Home computers just turned out not to be about math or complex programs.
65. Albert Charpentier, interview by Jesper Juul, April 22, 2022.
66. Andy Finkel, interview by Jesper Juul, April 22, 2022.
67. On a modern computer, we typically work with 24-bit color where we can decide the red, green, and blue intensity of each pixel on a scale from 0 (darkest) to 255 (brightest), so we use 1 byte for each of these colors, giving 3 bytes per pixel. To a modern reader this can appear simple—is this not just like a Photoshop document containing such an image? The problem is that this image would take 320-by-200-by-3 bytes of memory, (roughly) 192 KB of RAM, three times the C64 memory capacity.

68. In the VIC-II chip, sprites make up the bulk of the chip, indicating how important they—and arcade games—were to the designers (Perry and Wallich, “Design Case History”).
69. Tramiel, “Creating PETSCII.”
70. Each character can have its own color, but in bitmap mode, the colors for each 8-by-8 or 4-by-8 character area can be selected individually.
71. The first image in figure 0.16 shows the sixteen colors of the machine as it looks on a modern emulator, followed by a modern emulator mimicking the colors on an old TV. The latter is relevant because in practice the image was never as clear as the first image. Due to various issues in the chip and television standards, colors were often muted, and the screen was marred by artifacts like the minor dark vertical lines (“jailbars”) in the second image. This presents a dilemma for a book author: Should I use modern technology to show crisp colors, or should I replicate the blurry images of an old TV? Because I am interested in explaining how the C64 works, I have chosen to show clean, modern images. This is not historically correct, but they make it easier to see what is going on, and they reflect how C64 images look today.
72. Commodore, *Commodore 64 User’s Guide*, 57.
73. Herd, Charpentier, and Edwards, “Commodore VIC I and VIC II Chips.”
74. After every eighth pixel, the entire character screen must be shifted one character in programming.
75. Varga, “Interview with Bob Yannes”; Perry and Wallich, “Design Case History.”
76. Perry and Wallich, “Design Case History.”
77. Vintage Computer Federation, “Commodore Employees Panel.”
78. Ahl, “Incomplete History of Computer Games 1940–1990.”
79. Juul, “The Game, the Player, the World: Looking for a Heart of Gameness.”
80. Haddon and Skinner, “The Enigma of the Micro,” 443.
81. The Apple II was also designed for games, specifically *Breakout*, but this was before the development of the classic arcade game formula of quickly moving objects against a static or scrolling background, synchronized to the raster beam, and it is unsurprising that the Apple II lacks these features. Only the short-lived and home-oriented IBM PCjr (1984) included the game-oriented graphics features that IBM’s “serious” PC machines did not have.
82. Webster, “Microcomputer Color Graphics—Observations.”
83. Halfhill, “The MS-DOS Invasion: IBM Compatibles Are Coming Home.”
84. Bagnall, *Commodore*, 2019.
85. Natale and Balbi, “Media and the Imaginary in History”; Uricchio, “Television’s First Seventy-Five Years.”
86. I am referring to the conception of interpretative flexibility that emphasizes a closure where interpretative flexibility stops. See Doherty, Coombs, and Loan-Clarke, “A Re-Conceptualization of the Interpretive Flexibility of Information Technologies.”
87. Commodore, “VIC-20 Job Interview TV Commercial.”
88. Juul and Carney, “Would You Like Games with That Computer?”
89. Kidder, *The Soul of a New Machine*.
90. Custodio, *Who Are You?*
91. Nooney, *The Apple II Age*.
92. Gazzard, *Now the Chips Are Down*.
93. Maher, *The Future Was Here*; Montfort and Bogost, *Racing the Beam*.
94. Mailland and Driscoll, *Minitel*.

95. Ames, *The Charisma Machine*.
96. Natale and Balbi, “Media and the Imaginary in History.”

Chapter 1

1. Haddon and Skinner, “The Enigma of the Micro.”
2. Pollack, “The Home Computer Arrives.”
3. Cf. Kluitenberg, “On the Archaeology of Imaginary Media.”
4. That fantasies precede and inspire actual technology was not lost on the advertising professionals who promoted computers in the early 1980s, as they made the computer–science-fiction connection only slightly ham-fistedly. Isaac Asimov, author of *Foundation* and other science fiction classics, advertised the 1980 TRS-80 pocket computer, saying that a pocket computer used to be science fiction but was now available for purchase (RadioShack, “A Few Years Ago”). William Shatner of Star Trek fame made a similar case for the VIC-20, “A computer like this would have been science fiction a few years ago” (Commodore, “The First Honest-to-Goodness Full Color Computer”).
5. Peddle, “Lecture and Presentation of the Victor 9000 Computer.”
6. Light, “When Computers Were Women.”
7. Lancaster, “TV Typewriter.”
8. Charpentier and Herd, “40th Anniversary of the Commodore 64.”
9. Zenit Artivisive, Docabout, “Programma 101- Memory of Future,” 2010.
10. Stein, “Domesticity, Gender and the 1977 Apple II Personal Computer,” 196.
11. Atkinson, “The Best Laid Plans of Mice and Men,” n. 43.
12. Hiltzik, *Dealers of Lightning*, 257–276.
13. UK Commodore manager David Pleasance reports of a failed attempt at convincing Commodore president Mehdi Ali to create an international advertising strategy. (Pleasance, *Commodore The Inside Story*, loc. 1321.)
14. Since ads were uploaded by individual users, the sample likely favors large territories and fondly remembered ads.
15. Commodore, “Commodore 64—Are You Keeping Up?”
16. Commodore, “Are You Keeping Up with the Commodore (Family Pack).”
17. Tomczyk, “Preliminary Recommendations on the VIC Product Line,” June 20, 1980.
18. Michael S. Tomczyk, interview by Jesper Juul, September 2, 2023.
19. Haddon, “Electronic and Computer Games, the History of an Interactive Medium,” 210.
20. Haddon, 273.
21. Commodore, “Commodore 64 Italian Commercial.”
22. Pleasance, *Commodore The Inside Story*, loc. 674.
23. Commodore, “We Promise.”
24. Stein, “Domesticity, Gender and the 1977 Apple II Personal Computer,” 205.
25. Commodore, “Wer Entführt.”
26. Jesper Christensen, interview by Jesper Juul, August 8, 2022.
27. Fassonne, “Cammelli and Attack of the Mutant Camels”; Pleasance, *Commodore The Inside Story*, loc. 341.
28. Bagnall, *On the Edge*, 462.
29. Bagnall, *Commodore*, 2019, loc. 8641.
30. Rankin, *A People’s History of Computing in the United States*.

31. Rankin, 22.
32. Montfort et al., *10 PRINT CHR$(205.5+RND(1)); : GOTO 10*, 158.
33. Rankin, *A People's History of Computing in the United States*, 1.
34. Rankin, 29.
35. Dijkstra, "Go To Statement Considered Harmful."
36. Kemeny, *Man and the Computer*, 35.
37. Gates, "An Open Letter to Hobbyists."
38. "Bill Gates Interview."
39. According to Commodore history, Microsoft offered BASIC to Commodore for $3 per machine, but Jack Tramiel, stating "I'm already married," purchased the nonexclusive rights up front for a $25,000 lump sum (Bagnall, *On the Edge*, 81). This became a spectacularly good deal for Commodore considering the tens of millions of machines Commodore eventually sold and runs counter to stories of Microsoft's legendary business sense, such as when licensing MS-DOS to IBM.
40. Bagnall, *On the Edge*, 397.
41. Ahl, *BASIC Computer Games*, 1978; Ahl, *BASIC Computer Games*, 1973.
42. Ahl, *BASIC Computer Games*, 1978, 106.
43. Tramiel, "An Inexcusable Failure."
44. Nooney, *The Apple II Age*.
45. Andy Finkel, interview by Jesper Juul, April 22, 2022.
46. McMurtrie, "Gail Wellington."
47. McMurtrie, "The Commodore Clipper (and Other Tales of the Summer 1983 CES)."
48. Commodore Computer, "Commodore 64 Software."
49. Silveria, "Spreadsheets for the C-64."
50. Thissen, "Düsseldorf."
51. Brannon, "SpeedScript: Word Processor for The Commodore 64 And VIC-20."
52. Halfhill, "Word Processing in The Home."
53. Commodore, *Commodore 64 User's Guide*.
54. Montfort et al., *10 PRINT CHR$(205.5+RND(1)); : GOTO 10*.
55. Montfort et al., 171–172.
56. Wszola, "Hardware Review: Commodore 64."
57. Bötel, "Bundesliga-Manager."
58. Vose, "Software Review: True BASIC."
59. Benschop, "The history of Basicode."

Chapter 2

1. Wilkins and Kean, *Let's Go Dizzy: The Story of the Oliver Twins*.
2. Newman, "Cuthbert Goes Cloning."
3. "Investment Package."
4. Ken Williams of Sierra claims that their 1980 *Frogger* for the Apple II was the "first arcade game officially licensed for home computers" (Williams, *Not All Fairy Tales Have Happy Endings*, 396); Tommervik, "The Great Arcade / Computer Controversy. Part 1: The Publishers and the Pirates."
5. Der Spiegel, "Wie eine Epidemie."
6. Webster, "Microcomputer Color Graphics—Observations," 408.
7. Anderson, "Welcome to the First Issue of Zzap!64."
8. Kirkpatrick, *The Formation of Gaming Culture*.
9. Kirkpatrick, "Constitutive Tensions of Gaming's Field."
10. Kirkpatrick, *The Formation of Gaming Culture*, chap. 1.
11. Kirkpatrick, "Constitutive Tensions of Gaming's Field."
12. Pleasance, *Commodore The Inside Story*, loc. 674.
13. MobyGames, "About." I use MobyGames because it allows for comparison between platforms. MobyGames focuses on published titles, and its 6,200 C64 games can be compared to the GameBase64 database program, which at the time of writing counts 25,000 unique C64 game titles, including type-in programs and unpublished hobbyist games.
14. Commodore International, "Commodore International Limited Financial Statements," 3–4.
15. Wolf, *Before the Crash*.
16. Barclay, "Home Computers Soar, Video Games Uncertain."
17. Andy Finkel, interview by Jesper Juul, April 22, 2022.
18. Donovan, *Replay*, 103.
19. Nintendo, "Nintendo Co., Ltd. Consolidated Sales Transition by Region."
20. A small exception is Sweden, where a European version of the original Famicom was almost launched in 1985 (Bjarneby, "Historien om Bergsala—20 år med Nintendo"). The NES did also make some inroads (Gerli, "A Discussion on the European Gaming Market in the 80s").
21. "Segas Sell Better than Nintendos—Official!"
22. "Market Size and Market Shares."
23. Wilkins, *The History of Ocean Software*, 47–48.
24. "Top Twenty."
25. "HOTLINE: Top 30 Im Mai."
26. "HOTWARE: Software Best Sellers," November 1985.
27. "HOTWARE: Software Best Sellers," September 1986; "HOTWARE: Software Best Sellers," December 1986.
28. UK Interceptor Software did publish their first games in 1981 for the VIC-20.
29. In multiplayer *Wizard of Wor*, there is both a blue and a yellow astronaut, yet the sprite set contains only one astronaut. C64 sprites and characters contain no color information by themselves, and actual colors must be assigned by the programming. The maze, text, extra astronauts, and score information are all set to their color using the color memory. In the early 1980s, this was a beneficial choice as limited memory made it advantageous to separate the

information about pixels from the palette, allowing an image like the astronaut to be reused with different colors (a “palette swap”).

30. McMurtrie, “Building Wizard of Wor.” This kind of “cross-development” is useful because it allows you to keep the programming environment running on one computer, while the developed program can use the full memory of the C64, but such a setup only became available to smaller developers much later.
31. Tomczyk, *The Home Computer Wars*, 228–229.
32. Andy Finkel, interview by Jesper Juul, April 22, 2022.
33. In C64 assembly language, numbers are often written in hexadecimal notation (sixteen digits, counting 0123456789ABCDEF) signified by the $ sign. For example, the 64 KB of the C64 is 65,536 bytes, but in hexadecimal, this is more conveniently expressed as $10000, and 53,280 is $D020. It is not necessary to do or understand hexadecimal math, just note that this is a typical and convenient way of writing numbers when dealing with machine language.
34. 6502 machine language makes us do things differently than if we program in BASIC. To fill the screen with hearts in BASIC, the program would look like this.

```
10 FOR I=1024 TO 2023
20 POKE I,65
30 NEXT I
```

But in 6502 machine language, we do not immediately have variables larger than 255, and we would therefore rarely translate the BASIC program into a loop that counts from 1024 to 2023; we reuse a counter instead. A typical way to fill the character screen with hearts is to have only one loop, but store four values for every step:

```
LDX #0. ;LoaD X with 0
LDA #65. ;LoaD 65 into A (the heart character)
loop ;This label indicates the start of the loop
STA 1024,x ;Store A in screen memory at 1024+x
STA 1024+256,x ;Store A
STA 1024+512,x ;Store A
;Store A—but subtract 24 to stay in screen
memory
STA 1024+768–24,x
INX ;Increase X
BNE loop ;As long as X is not zero

again, continue the loop
```

35. “Software Reviews: Impossible Mission.”
36. Bevan, “The History of . . . Impossible Mission,” 45.
37. Bevan, 47.

38. We may at first get the impression that *Summer Games* has transcended the eight sprite limit, but closer inspection reveals that the doves are separated vertically from the runner and flame; they move in two separate sections of the screen. Disassembling the *Summer Games* program, it turns out at the top of the screen, the program sets the eight sprites to be the eight doves. After the doves have been displayed, the program then sets the eight sprites to be the runner and the fire. Incidentally, this is exactly the technique that the developer-only *Commodore 64 Software Development Kit* uses in the *Multiple Sprite Interrupt Routine* example (Commodore, *Commodore 64 Software Development Kit*). Disassembling the *Summer Games* program shows that it is not the same code, but the same principle. As a large developer, Epyx likely had access to the kit and the example. Why does the opening ceremony with its sixteen sprites work for the viewer? It is not just a technical trick but a content trick. It makes sense for a runner to be on the ground and doves to be in the air. Ceremonial white doves *are* usually released from the ground though, but here they are conveniently released outside our view, as to only appear on the top third of the screen, making it work with two sets of eight sprites.
39. WWS/RWE + J. S., "'Illegal' Presents the OLD 64 LEGENDS!"
40. Montfort and Bogost, *Racing the Beam: The Atari Video Computer System*, chap. 7.
41. Donovan, *Replay*.
42. Megler, "There and Back Again: A Case History of Writing 'The Hobbit.'"
43. In my book *Half-Real*, I imprecisely used *The Hobbit* as an example of a *game of progression*, where the player must perform a preset sequence of actions to complete the game. While the game *can* be completed with a fixed sequence of commands, it can be played in many ways and is much more dynamic and emergent than other adventure games (Juul, *Half-Real*, 67–71).
44. Montfort, *Twisty Little Passages*, 126–127.
45. *The Hitchhiker's Guide to the Galaxy*'s footnote system expresses the humor of Douglas Adams well by presenting bureaucratic systems and formal conventions, such as footnotes, as absurd. As the game says, it *is* fun reading all the footnotes.
46. Wade, *Playback—A Genealogy of 1980s British Videogames*, 129.
47. "Matthew Smith Interview Manic Miner ZX Spectrum."
48. Juul, *Half-Real*.
49. "Penguins Make Life Perilous!!—Manic Miner"; "Manic Miner Review."
50. Caulfield and Caulfield, *Jet Set Willy Making Of*.
51. Hardisty, *A Gremlin in the Works: 1983–1989*, 44–53.
52. Crowther, *Wanted: Monty Mole*.
53. Dekel, "Wanted"; Hardisty, *A Gremlin in the Works: 1983–1989*, 51.
54. Jenkins, "Gremlin Graphics Presents."
55. Montfort and Bogost, *Racing the Beam: The Atari Video Computer System*, chap. 3; Oliveira et al., "A Framework for Metroidvania Games."
56. Allen, "Interview with Dennis Caswell."
57. SNC, "Son of Blagger Review."
58. Anthropy and Clark, *A Game Design Vocabulary*, chap. 1.
59. "Monty on the Run Review."
60. Hardisty, *A Gremlin in the Works: 1983–1989*, 161. The visual abilities of the ZX Spectrum are basically equivalent to the C64 hi-res mode (slightly smaller, at 256-by-192 pixels) where the colors of each 8-by-8 cell can be defined independently. In the absence of sprites and scrolling facilities, a Spectrum developer must update all pixels individually, meaning that ZX Spectrum

games only scroll under limited circumstances and that moving objects often flicker. The absence of sprites on the ZX Spectrum also means that you cannot have overlapping objects of different colors, leading to the infamous “colour clash,” where part of an object must change color when passing through another.

61. Hubbard, “Golden Days of Computer Game Music (Part 1).”
62. Banks, “The Wrong Tool for the Right Job,” 71.
63. Hardisty, *A Gremlin in the Works: 1983–1989*, 36.
64. Hardisty, 164–165.
65. Hubbard, “Rob Hubbard.”
66. McSweeney, “Rob Hubbard’s Music: Disassembled, Commented and Explained.”
67. Morrison, “8-Bit Symphony and Rob Hubbard.”
68. Driscoll and Diaz, “Endless Loop: A Brief History of Chiptunes.”
69. McSweeney, “Rob Hubbard’s Music: Disassembled, Commented and Explained.”
70. Hubbard, “Interview by 6581SID (KDVS).”
71. Collins, “‘Loops and Bloops’—Music of the Commodore 64 Games.”
72. A less common observation is that C64 music feels like it has a limited number of tempos and is often fixed at a common up-tempo beat. If we measure the tempo of popular C64 music on European machines, the most common tempo is exactly 125 beats per minute (BPM). Why would that be? Because C64 music programs typically update once per frame of the video signal, meaning fifty times (PAL) or sixty times (NTSC). Consider a 120 BPM song. Given that there will be two beats per second, each beat will be a half-second long, corresponding to twenty-five frames per beat on a European PAL machine. But this is a problem, because twenty-five does not divide by two, and we therefore cannot play any shorter notes (such as 1/8th notes), rendering this tempo impractical. (We could do a 1/8th note on an NTSC machine though, as 30 (60/2) divides by two.) What about the *Monty on the Run* music? This is 94 BPM (93.75 to be exact), which turns out to correspond to thirty-two frames for each beat, allowing the playing of 1/8th, 1/16th, 1/32th, and ever shorter notes. Compare this to the *Robocop 3* music running at the more common 125 BPM, which translates to twenty-four frames per beat, which can be subdivided into shorter notes, as well as to 1/3rds, giving triplets. In short, the composition choice is whether the shortest note should correspond to one, three, or five frames. (Since both 4 and 2 can be subdivided into 1, these are the same.) A quick survey shows that this corresponds to the three main tempos in C64 music, with 125 BPM being the dominant tempo.

    - Shortest note is one frame: 94 BPM (*Monty on the Run*, *Last Ninja 2*, *Ocean Loader 2*, *Green Beret*)
    - Shortest note is three frames: 125 BPM (*Robocop 3*, *Last Ninja*, *Cybernoid*, *Commando*, *IK+*, *Great Giana Sisters*, *Thing on a Spring*, *Auf Wiedersehen Monty*, *Crazy Comets*, *Sanxion* loading music)
    - Shortest note is five frames: 150 BPM (*Delta*)

    A composer could potentially update the music player twice per frame for more tempo flexibility, but this was rare.
73. Hülsbeck, “Musik . . . Wie Noch Nie.”
74. Carlsson, “Power Users and Retro Puppets,” 19.
75. “Interview: Peter Liepa.”
76. M. W., “Boulder Dash CBM 64 £8.95.”

77. “Interview: Peter Liepa.”
78. Jones, *The Commodore 64 Book: 30th Anniversary Special*, 61.
79. Braybrook, “Birth of a Paradroid. Part I.”
80. Anderson, “Crowther: Super Being a Superstar!”
81. Consalvo, *Cheating*, 95–98.
82. Commodore, *Commodore 64 Programmer’s Reference Guide*.
83. Wilkins, *The History of Ocean Software*, 45–47.
84. Even if you have not programmed 3D, it is probably obvious that 3D graphics require math and trigonometry. The 6502 can feel like it was deliberately designed not to do this kind of math, and that is not far off. How would you do this anyway? One word: tables. We can perform multiplication by looking up the logarithm of two numbers in a table, adding the two values, and looking up the reverse logarithm. Similarly, we can have tables for sine and cosine values.
85. Juul, “Without a Goal: On Open and Expressive Games.”
86. Minter, “Psychedelia—History.”
87. Gingold, *Building SimCity: How to Put the World in a Machine*, chap. 7.
88. Donovan, *Replay*, 187.
89. Gingold, *Building SimCity: How to Put the World in a Machine*, chap. 6.
90. Gingold, chap. 8.
91. Maher, “Will Wright’s City in a Box.”
92. Gingold, *Building SimCity: How to Put the World in a Machine*, chap. 6.
93. “SimCity Review.”
94. “Talking with Manfred Trenz.”
95. Maclean and Ant & Nic’s Cabin Fever, “Creating International Karate & IK+.”
96. Reggio, *Koyaanisqatsi*.
97. Kirchesch, “Chris Hülsbeck Über Turrican, Great Giana Sisters.”
98. It is a complex picture, but if we look at the bigger magazines, US *Ahoy!* shut down in 1989, *RUN* had its last issue in 1992, and *Compute’s Gazette* ceased to be a standalone print magazine in 1990, whereas UK *Zzap!64* continued to 1994 (as *Commodore Force*), UK *Commodore Format* continued until 1995, and German *64’er* until 1996. US *Commodore World Magazine* continued until 1999 though.
99. Noah Falstein, correspondence with the author, June 2023.
100. “The February Charts!”; “Hitline in Februar.”
101. Email exchange with Gleb Albert, January 2024, based on his in-progress work on the history of the cracking scene, *Ökonomien des Kopierens*.
102. Nooney, *The Apple II Age*, chap. 5.
103. Bagnall, *On the Edge*, 506.
104. Wilkins and Kean, *The Story of US Gold*, 203.
105. Codetapper, “California Games C64 Copy Protection.”
106. Facebook discussion with the author, June 2023.
107. Email exchange with Gleb Albert, January 2024, based on his in-progress work on the history of the cracking scene, *Ökonomien des Kopierens*.
108. Some US developers, like Jordan Mechner, were also in their teens, but the overall pattern holds.

109. Albert, “New Scenes, New Markets.”
110. “Ny Lag Stoppar Pirat-Kopiering.”
111. “Madonna Cracking Group Party”; Mr. Cursor, “Reactions to the MCG Party.”
112. Swyx (Triangle), interview by Jesper Juul, December 10, 2021.
113. Jesper Christensen, interview by Jesper Juul, August 8, 2022.
114. Fassonne, “Cammelli and Attack of the Mutant Camels.”
115. Gerli, “A Swashbuckling Tale of Italian Software Piracy—1983–1993.”
116. Der Spiegel, “Eng und holzig.”
117. “Raubkopieren.”
118. Beiler, “Haie Und Kleine Fische.”
119. Beiler.
120. “Neue Tauschpartner!!”
121. WWS/RWE + J. S., “‘Illegal’ Presents the OLD 64 LEGENDS!”
122. Wasiak, “‘Illegal Guys’: A History of Digital Subcultures,” 261.
123. Kirkpatrick, *The Formation of Gaming Culture*.
124. Randall, “The Last Crusade.”
125. Dietz, “An Examination of Violence and Gender Role Portrayals.”
126. Garda and Grabarczyk, “But Our Princess Is on Another Platform . . .”
127. Juul, *Handmade Pixels*.
128. Morningstar and Farmer, “The Lessons of Lucasfilm’s Habitat”; Morabito, “Enter the Online World of LucasFilm.”
129. Hammond, “GTA Origins.”

Chapter 3

1. Bennett, *Vibrant Matter*, viii, 21.
2. Rignall, “The Compunet Page.”
3. “Interview with Joost Honig (Honey).”
4. Polgár, *Freax: The Brief History of the Computer Demoscene*; Tasajärvi, *Demoscene*; Botz, *Kunst, Code und Maschine*.
5. Reunanen and Silvast, “Demoscene Platforms”; Hege, “The Gathering–Computer Parties as Means for Gender Inclusion.”
6. Hartmann, *Digital Art Natives*, chap. 6.
7. Tasajärvi, *Demoscene*; Tyni and Sotamaa, “Assembling a Game Development Scene?”; Saarikoski and Suominen, “Computer Hobbyists and the Gaming Industry in Finland”; neogames, “The Game Industry of Finland Report 2020.”
8. The demoscene–industry connection was later readdressed by Jørgensen, Sandqvist, and Sotamaa, who acknowledged that “seeing demoscene simply as a training ground or ‘preschool’ for the game industry would be a crude simplification” (Jørgensen, Sandqvist, and Sotamaa, “From Hobbyists to Entrepreneurs”).
9. Reunanen, “Times of Change in the Demoscene.”
10. Reunanen, “How Those Crackers Became Us Demosceners,” 29–30.
11. Botz, *Kunst, Code und Maschine*.
12. Wasiak, “‘Illegal Guys’: A History of Digital Subcultures”; Albert, “Subkultur, Piraterie und neue Märkte”; Albert, “New Scenes, New Markets.”
13. Botz, *Kunst, Code und Maschine*, 36–40.
14. Raymond, *The New Hacker’s Dictionary*.
15. Levy, *Hackers*.
16. Lyons, “A Short History of the Demo . . .”; Tasajärvi, *Demoscene*.
17. Reunanen, “How Those Crackers Became Us Demosceners.”
18. Chix (Triangle), interview by Jesper Juul, January 19, 2024.
19. Trap (Bonzai), interview by Jesper Juul, January 4, 2024.
20. “Interview with Joost Honig (Honey).”
21. Wasiak, “Computer Dealer Demos”; Hastik and Steinmetz, “Demoscene Computer Artists and Community.”
22. Wasiak.
23. Hartmann, “Animation in the Demoscene.”
24. Botz, *Kunst, Code und Maschine*, 22.
25. Commodore 64 Scene Database, “CSDb Help / Documentation”; “The C-64 Scene Database”; Rahm-Skågeby and Carlsson, “The Archive and the Scene.”
26. Albert, “Got Papers?”
27. Therrien, “Video Games Caught Up in History.”
28. Intangible Cultural Heritage, “Demoscene, Musical Saw Playing and Horsemanship of the Roma—12 New Elements Inscribed on the National Inventory of Living Heritage”; Deutsche UNESCO-Kommission, “Demoszene—Kultur der digitalen Echtzeit-Animationen”; Lichnerowicz, “Komitet Społeczny Kronika Polskiej Demosceny”; Kenniscentrum Immaterieel,

"215e bijschrijving in Inventaris Immaterieel Erfgoed Nederland"; Bundesamt für Kultur, "Die Liste Der Lebendigen Traditionen Der Schweiz Wächst Weiter."

29. Pouët.net, "pouët.net 2.0 API."
30. "AEK Copy-Party 1986."
31. Trap (Bonzai), interview by Jesper Juul, January 4, 2024.
32. "This Is How We Play It."
33. "Dominators/Upfront/Trilogy Party 1989."
34. Fuller, *Media Ecologies*, 15.
35. Driscoll, *The Modem World*, 72.
36. Moleman, "Stamps Back."
37. Einstein (Upfront), interview by Jesper Juul, April 14, 2023.
38. The Pirates Crew, "Pirates #08."
39. "Coococ #10."
40. "Fatal News #04."
41. Babygang, "BYG #2."
42. Success, "Lethal News #1."
43. Bertelsen and Nelleman, "Demo Time: I Went to a Copy Party."
44. "Dominators/Upfront/Trilogy Party 1989 Results."
45. Tasajärvi, *Demoscene*, 25.
46. Ninja / The Dreams, "Behind the Scenes of a C64 Demo"; Polgár, *Freax: The Brief History of the Computer Demoscene*, chap. 6.
47. Tasajärvi, *Demoscene*, 65–68.
48. Botz, *Kunst, Code und Maschine*, 247–248.
49. Similarly, the Triangle logo that scrolls in from the border appears to freely move around the entire width of the screen (figure 3.11), but the sideborder can be shut off once the logo is centered and more action needs to happen on screen.
50. Baal. "The News."
51. "Coococ #10."
52. Einstein (Upfront), interview by Jesper Juul, April 14, 2023.
53. Swyx (Triangle), interview by Jesper Juul, December 10, 2021.
54. Swyx (Triangle).
55. Chix (Triangle), interview by Jesper Juul, January 19, 2024.
56. Hege, "The Gathering–Computer Parties as Means for Gender Inclusion."
57. Swyx (Triangle), interview by Jesper Juul, December 10, 2021.
58. Johan Grip, interview by Jesper Juul, May 18, 2022.
59. Albert Charpentier, interview by Jesper Juul, April 22, 2022.
60. The 6510 CPU and the VIC-II graphics chip in the C64 are locked in fierce competition for access to the RAM chips on the lower left (figure 0.10) of the circuit board. The memory "bus"—the lines that move data around—works at 1 MHz, divided into a "low" and "high" phase. At the low phase, the VIC-II graphics chip has access to RAM, and at the high phase, the 6510 gets access except when blocked by the VIC-II, which happens in two situations:

1. At the top of every row of text, the VIC-II must fetch the list of *which* characters are to be displayed in the current row, and therefore blocks the 6510 from RAM access for most of the line, known as *badlines*. This made it complicated to have sideborder sprites while showing text.
2. For every sprite, the VIC-II chip must fetch an additional 3 bytes per line and block the 6510. When the top of a character row is covered in sprites, there is thus almost no time left for the 6510.

But doesn't the VIC-II need to read—at each badline—an additional 40 bytes to determine the color of each character on the screen? The C64 is actually a 64.5 KB computer, because the color memory is its own 0.5 KB RAM chip of the fast SRAM type, which the VIC-II can read independently of the main RAM, and without blocking the CPU. Without that extra 0.5 KB of RAM, the C64 would not be able to work.

61. Wasiak, "'Amis and Euros.' Software Import and Contacts Between European and American Cracking Scenes."
62. Montfort and Bogost, *Racing the Beam: The Atari Video Computer System*.
63. Commodore, *Commodore 64 Programmer's Reference Guide*, 131.
64. Since the 6510 has no trigonometric functions, the circle movement is created using a precalculated table (or list) of cosine and sine values for the x and y positions of the sprites. In the 1980s, this table would be created with a BASIC program.
65. The VIC-II chip activates the top/bottom border by waiting for a specific line. In 24-line mode, it waits for line 248 before beginning to show the border. In regular 25-line mode, it waits for line 252, four lines below. If we set the chip to 25-line mode—making it wait for line 252—but switch to 24-line mode on line 249, it will wait instead for line 248 which has already passed and therefore never comes. The VIC-II thus never realizes that it should start drawing the border and instead shows what is underneath—the background color and sprites (figure 0.13). The same happens in the sideborder where switching between 38 and 40 columns makes the VIC-II chip forget to draw the border.

    The top/bottom border trick requires activating 24-line mode once per frame (50 or 60 times per second), but the sideborder requires activating 38-column mode at the exact horizontal position on *every line*, more than 15,000 times per second, require both more precise timing, and tying up the 6510 CPU most of the time.
66. Because sprites require the VIC-II chip to access memory, the VIC-II pauses the CPU a few cycles on every line. Thus, moving one sprite a single vertical pixel is enough to disturb the timing of a program and cause a sideborder routine not to work. So, at first it was not understood how to move sprites vertically while having an open sideborder. The *DYSP*, or *Different Y Sprite Positions*, trick was first published as a joke, claiming to have invented a new technique but in practice just animating sprites that were standing still. Later, this became a real effect, as programmers learned to compensate for how sprites change the timing on every line.
67. Lft, "Safe VSP."
68. "Graphics Modes."
69. Trap (Bonzai), interview by Jesper Juul, January 4, 2024.
70. Einstein (Upfront), interview by Jesper Juul, April 14, 2023.
71. "Dexion X-Mas Conference 90." At the time, Denmark had lax and rarely enforced laws concerning underage drinking.
72. Sørensen, "Udmattede Unge EDB-Entusiaster."
73. Edited for clarity. "The Party 2002," https://csdb.dk/event/?id=514.

74. Juul, *Handmade Pixels*, 71.
75. Einstein (Upfront), interview by Jesper Juul, April 14, 2023.
76. Rothstein, “Corruption and Social Trust,” 1010; Henriksen, Koch-Nielsen, and Rosdahl, “Formal and Informal Volunteering in a Nordic Context,” 196.
77. Carlsson, “The Forgotten Pioneers,” 18.
78. Juul and Carney, “Would You Like Games with That Computer?”
79. Anderson, “Welcome to the First Issue of Zzap!64.”
80. Liddon, “The Genius of Amplified Light.”
81. Liddon, “Transporting Sprites Across the Border.”
82. Kelly, “Impossible Scroll”; Gehrmann, “Hyperscreen.”
83. “The Charley Knight Mega-Muzix Routine.”
84. Kirkpatrick, *The Formation of Gaming Culture*, chap. 3.
85. Klein, “Profi-Corner”; Klein, “Assembler Corner: Zeichensatz & Scrolling.”
86. Burkert, “Camelot-Weinachts-Copy-Party”; Hartung, “Szene Inside.”
87. Matteo Bittanti, interview by Jesper Juul, June 2023.
88. Kirkpatrick, *The Formation of Gaming Culture*, 127.
89. Swalwell, *Homebrew Gaming and the Beginnings of Vernacular Digitality*.
90. bwack, “The VIC II Die—Explaining the Border Removal Trick (C64).”

Chapter 4

1. Halfhill, “RIP Commodore.”
2. Halfhill, “A Turning Point for Atari.”
3. Lock and Halfhill, “Editor’s Notes.”
4. Engelbart, “The Strategic Pursuit of Collective IQ.”
5. Commodore had attempted to create a C64 graphical user interface already with the largely unsuccessful *Magic Desk* (1983), using the game-oriented joystick and sprite facilities to create a pointer and icons on a “desktop” which was basically a menu for simple office programs.
6. Commodore International, “Redesign of the Popular C64.”
7. Schramm and Schneider, “Der Neue,” 43.
8. Noel, “GEOS: The Quiet Revolution.”
9. The regular joystick input pins were designed for digital devices where a switch is either on or off, but a mouse is an analog device with a need to measure how far the mouse has moved. To support paddles, the joystick port had analog inputs connected to the sound chip, and the Commodore 1351 mouse connected to these analog inputs.
10. Elko, “Editor’s Notes.”
11. Absmeier, “Heimverleger.”
12. Mast, “Final Cartridge III Manual.”
13. “Final Cartridge III: All New, Now with Windows!”
14. “Defender of the Crown Review,” *INFO Magazine*.
15. Cinemaware, “Defender of the Crown Instruction Manual.”
16. “Defender of the Crown Review,” *Zzap!64*.
17. Commodore Force, “Lemmings: An EXCLUSIVE Demo from Psygnosis!”
18. Bannon, “The Making of Lemmings (1 of 3).”
19. Maher, *The Future Was Here*, chap. 2.
20. Drawing 3D objects on the Amiga is efficient as the Amiga has hardware for drawing lines and filling shapes. 3D calculations are easier as the new CPU has larger registers and can perform multiplication. The ample RAM allows for more table-based effects.
21. Reunanen, “Computer Demos—What Makes Them Tick?” 83; Botz, *Kunst, Code und Maschine*, 241–249.

Chapter 5

1. Reunanen, Heikkinen, and Carlsson, “PETSCII—A Character Set and a Creative Platform,” 31; Silicon Limited, Success + The Ruling Company, and Xenon, “X’2023.”
2. Samar Crew, “$11 Music Compo”; Banks, “The Wrong Tool for the Right Job,” 76.
3. Price, “Editorial.”
4. Bordwell, *On the History of Film Style*.
5. Juul, *Handmade Pixels*, chap. 2.
6. Meyers, “Keys of Fury—Type in Beyond the Scrolling Horizon.”
7. Raquel Meyers, interview by Jesper Juul, August 17, 2023.
8. Street Tuff, “ECM Mode Demoeffect Competition.”
9. Åkesson, “Two C64s Plus a Pile of Floppy Disks Equals One Accordion.”
10. Herd and Morabito, *Back into the Storm*, 28.
11. Albert Charpentier, interview by Jesper Juul, April 22, 2022.
12. Stachniak, “Red Clones.”
13. Retrospective, “EVO64 is a completely modern, yet still fully authentic Commodore 64 remake, with tons of new capabilities.”
14. Džin, “ARMSID”; Rossi, “VIC-II Kawari.”
15. Retro Recipes, “All New Commodore 64 Using Ultimate 64 & LEGO Keycaps!”
16. Stephen Edwards, interview by Jesper Juul, May 25, 2022.
17. Grip, “Introducing the C128 Neo.”
18. Johan Grip, interview by Jesper Juul, May 18, 2022.
19. Juul and Norton, “Easy to Use and Incredibly Difficult.”
20. Magaudda and Minniti, “Retromedia-in-Practice.”
21. Magaudda and Minniti, 682.
22. hedning, “Gubbdata 10 Years.”
23. ZZAP69, “Gubbdata 2022.”
24. Pleasance, *Commodore The Inside Story*, loc. 3105.
25. Andrews, *TOO BLUE!*, loc. 102.
26. Halfhill, “RIP Commodore.”
27. Pleasance and Dickinson, *From Vultures to Vampires*.
28. Rankin, *A People’s History of Computing in the United States*, 11.
29. Black, “Usable and Useful,” 518.
30. Doherty, Coombs, and Loan-Clarke, “A Re-Conceptualization of the Interpretive Flexibility of Information Technologies,” 570–571; Skågeby and Rahm, “What Is Feminist Media Archaeology?” 2.
31. Botz, *Kunst, Code und Maschine*, 392; Mansoux et al., “Permacomputing Aesthetics.”

Bibliography

Absmeier, Albert. “Heimverleger.” *64’er Magazin*, January 1988.

“AEK Copy-Party 1986.” C-64 Scene Database, 1986. https://csdb.dk/event/?id=636.

Ahl, David H., ed. *BASIC Computer Games*. Digital Equipment Corporation, 1973.

———, ed. *BASIC Computer Games: Microcomputer Edition*. New York: Workman, 1978.

———. “Incomplete History of Computer Games 1940–1990.” Vintage Computer Festival East, October 9, 2021. https://www.youtube.com/watch?v=eXCSKPiGFwM.

Åkesson, Linus. “The Commodordion.” October 21, 2022. https://www.youtube.com/watch?v=EBCYvoC4muc.

———. “Two C64s Plus a Pile of Floppy Disks Equals One Accordion.” *IEEE Spectrum*, December 23, 2022. https://spectrum.ieee.org/two-commodore-64s-one-accordion.

Albert, Gleb J. “Got Papers? Preserve the Scene’s Material Heritage,” February 26, 2023. https://gotpapers.scene.org/.

———. “New Scenes, New Markets: The Global Expansion of the Cracking Scene, Late 1980s to Early 1990s.” *WiderScreen* 20, no. 1–2, 2020. http://widerscreen.fi/numerot/2020-2-3/new-scenes-new-markets-the-global-expansion-of-the-cracking-scene-late-1980s-to-early-1990s/.

———. *Ökonomien Des Kopierens, Kulturen Des „Knackens“. Softwarepiraterie Als Jugendkultur Im Heimcomputer-Zeitalter, 1980–1995*, Forthcoming.

———. “Subkultur, Piraterie und neue Märkte: Die transnationale Zirkulation von Heimcomputersoftware, 1986–1995.” In *Wege in die digitale Gesellschaft: Computernutzung in der Bundesrepublik 1955–1990*, ed. Frank Bösch and Martin Sabrow, 274–299. Göttingen, Germany: Wallstein Verlag, 2018.

Allen, Matt. “Interview with Dennis Caswell.” Mayhem UK Commodore 64 archive. Accessed January 18, 2024. http://www.mayhem64.co.uk/interview/caswell.htm.

Ames, Morgan G. *The Charisma Machine: The Life, Death, and Legacy of One Laptop per Child*. Cambridge, MA: MIT Press, 2019.

Anderson, Chris. “Crowther: Super Being a Superstar!” *ZZap!64*, May 1985.

———. “Welcome to the First Issue of Zzap!64.” *Zzap!64*, May 1985.

Andrews, Dennis. *TOO BLUE!: The IBM PC from an Acorn to a Renegade*. Levanto, Italy: Leaders Press, 2020.

Anthropy, Anna, and Naomi Clark. *A Game Design Vocabulary: Exploring the Foundational Principles Behind Good Game Design*. Upper Saddle River, NJ: Addison-Wesley Professional, 2014.

Arnold, Bruce Makoto. “Twenty-Two Columns of Lowbrow Revolution: The Commodore VIC-20 and the Beginning of the Home Computer Era.” *Journal in Humanities* 6, no. 1 (2017): 11–20.

Atkinson, Paul. “The Best Laid Plans of Mice and Men: The Computer Mouse in the History of Computing.” *Design Issues* 23, no. 3 (July 1, 2007): 46–61.

Baal. “The News.” *Pirates*, 1990.

Babygang. “BYG #2,” February 28, 1990. https://mirror.netcologne.de/scene.org/mags/byg/byg02-paper.pdf.

Bagnall, Brian. *Commodore: The Amiga Years*. Winnipeg, Manitoba: Variant Press, 2019.

———. *Commodore: The Final Years*. Winnipeg, Manitoba: Variant Press, 2019.

———. *On the Edge: The Spectacular Rise and Fall of Commodore*. 2nd ed. Winnipeg, Manitoba: Variant Press, 2010.

Banks, Tobias. “The Wrong Tool for the Right Job: Composition on 8-Bit Machines.” *INSAM Journal of Contemporary Music, Art and Technology* 2, no. 3 (2019): 69–81.

Bannon, Roy. “The Making of Lemmings (1 of 3).” *Commodore Force*, September 1993.

Barclay, Doloroes. “Home Computers Soar, Video Games Uncertain.” *Associated Press*, December 9, 1982.

Beiler, Harald. “Haie Und Kleine Fische.” *64’er Magazin*, October 1994.

Bennett, Jane. *Vibrant Matter: A Political Ecology of Things*. Durham, NC: Duke University Press, 2010.

Benschop, Lennart. “The history of Basicode.” *Hobbyscoop.nl* (blog), n.d. https://hobbyscoop.nl/the-history-of-basicode/.

Bertelsen, Rasmus, and Adam Nelleman. “Demo Time: I Went to a Copy Party.” *Det Nye Computer*, February 28, 1990.

“Best-Selling Desktop Computer.” Guinness World Records. Accessed May 6, 2022. https://www.guinnessworldrecords.com/world-records/72695-most-computer-sales.

Bevan, Mike. “The History of . . . Impossible Mission.” *Retro Gamer*, no. 122 (2013).

Bjarneby, Tobias. “Historien om Bergsala—20 år med Nintendo.” *IDG.se* (blog), September 29, 2006. https://www.idg.se/2.1085/1.77792/historien-om-bergsala—20-ar-med-nintendo.

Black, Michael L. “Usable and Useful: On the Origins of Transparent Design in Personal Computing.” *Science, Technology, & Human Values* 45, no. 3 (May 1, 2020): 515–537.

Bordwell, David. *On the History of Film Style*. Cambridge, MA: Harvard University Press, 1998.

Bötel, Jörg. “Bundesliga-Manager.” *Happy Computer*, July 1986.

Botz, Daniel. *Kunst, Code und Maschine: Die Ästhetik der Computer-Demoszene*. Bielefeld, Germany: transcript, 2011.

Brannon, Charles. “SpeedScript: Word Processor for The Commodore 64 And VIC-20.” *Compute!’s Gazette*, January 1984.

Braybrook, Andrew. “Birth of a Paradroid. Part I.” *Zzap!64*, July 1985.

Bundesamt für Kultur. “Die Liste Der Lebendigen Traditionen Der Schweiz Wächst Weiter,” August 22, 2023. https://www.bak.admin.ch/bak/de/home/aktuelles/nsb-news.msg-id-97387.html.

Burke, Martyn, dir. *Pirates of Silicon Valley*. Atlanta, GA: TNT, 1999.

Burkert, Jorn-Erik. “Camelot-Weinachts-Copy-Party.” *64’er Magazin*, January 1994.

bwack. “The VIC II Die—Explaining the Border Removal Trick (C64).” April 10, 2016. https://www.youtube.com/watch?v=nkf02t5tFek.

“The C-64 Scene Database.” C-64 Scene Database, 2023. https://csdb.dk/.

Carlsson, Anders. “Power Users and Retro Puppets—a Critical Study of the Methods and Motivations in Chipmusic.” Master’s thesis, Lund University, 2010. http://lup.lub.lu.se/student-papers/record/1662548.

———. “The Forgotten Pioneers of Creative Hacking and Social Networking—Introducing the Demoscene,” 2009.

Caulfield, Nicola, and Anthony Caulfield, dir. *Jet Set Willy Making Of*. Rubber-Keyed Wonder Backer Pack, 2022. https://vimeo.com/696626875/5cd984fb1f.

Ceruzzi, Paul E. *A History of Modern Computing*. 2nd ed. Cambridge, MA: MIT Press, 2003.

"The Charley Knight Mega-Muzix Routine." *ZZap!64*, October 1989.

Charpentier, Albert, and Bil Herd. "40th Anniversary of the Commodore 64." Presented at the Vintage Computer Festival Southeast, Atlanta, GA, May 15, 2023. https://www.youtube.com/watch?v=SFs6KmJQH3w.

Cinemaware. "Defender of the Crown Instruction Manual." Mindscape, 1986.

Clarke, Arthur C. *The Collected Stories of Arthur C. Clarke*. New York: RosettaBooks, 2016.

Codetapper. "California Games C64 Copy Protection." 2018. https://codetapper.com/c64/rants/california-games-c64-copy-protection/.

Collins, Karen. "'Loops and Bloops'—Music of the Commodore 64 Games." *Soundscapes.Info* 8 (February 2006).

Commodore. "Are You Keeping Up with the Commodore?" 1983. https://www.youtube.com/watch?v=5uaYHYs4ubw.

———. "Are You Keeping Up with the Commodore (Family Pack)." 1984. https://www.youtube.com/watch?v=wdYTYIiCdEY.

———. "Commodore 64 Italian Commercial." 1983. https://www.youtube.com/watch?v=7wylcEGgKWY.

———. *Commodore 64 Programmer's Reference Guide*. Commodore Business Machines, 1982.

———. "Commodore 64 Software," Commodore Computer, 1982.

———. *Commodore 64 Software Development Kit*. Commodore Business Machines, 1982. https://archive.org/details/commodore64softwaredevelopmentkit.

———. *Commodore 64 User's Guide*. Wayne, PA: Commodore Business Machines, 1982.

———. "The First Honest-to-Goodness Full Color Computer You Can Buy for Only $299.95," 1982.

———. "VIC-20 Job Interview TV Commercial." 1981. https://www.youtube.com/watch?v=BZvJzB7ezBY.

———. "We Promise You Won't Use the Commodore 64 More Than 24 Hours A Day." *Commodore Microcomputers*, January 1985.

———. "Wer Entführt Euch In Die Wunderwelt Der Wissenschaft?" *64'er Magazin*, April 1984.

Commodore International. "Commodore International Limited Financial Statements for the Worldwide Bank Group March 1986." March 1986.

———. "Redesign of the Popular C64 Bundled with the Graphic Environment Operating System (GEOS)." PR Newswire, June 2, 1986.

Commodore 64 Scene Database. "CSDb Help / Documentation." Accessed January 24, 2024. https://csdb.dk/.

Consalvo, Mia. *Cheating: Gaining Advantage in Videogames*. Cambridge, MA: MIT Press, 2007.

"Coococ #10." C-64 Scene Database, 1989. https://csdb.dk/release/?id=18313.

Cringely, Robert X. *Accidental Empires: How the Boys of Silicon Valley Make Their Millions, Battle Foreign Competition, and Still Can't Get a Date*. Reading, MA: Addison-Wesley, 1992.

———. *Accidental Empires: How the Boys of Silicon Valley Make Their Millions, Battle Foreign Competition, and Still Can't Get a Date*. Reprint edition. New York: Harper Business, 1996.

Crowther, Tony. *Wanted: Monty Mole*. Gremlin Graphics (C64), 1984.

Custodio, Alex. *Who Are You?: Nintendo's Game Boy Advance Platform*. Cambridge, MA: MIT Press, 2020.

De Bartolo, Dick, and Harry North. "The Computer Company President of the Year." *MAD Magazine*, October 1985.

"Defender of the Crown Review." *INFO Magazine*, January 1987.

"Defender of the Crown Review." *Zzap!64*, August 1987.

Dekel, Jesse. "Wanted: Monty Mole." *Organise Magazine* (blog), September 26, 2019. https://organisemagazine.org.uk/2019/09/26/wanted-monty-mole-rbg/.

Deutsche UNESCO-Kommission. "Demoszene—Kultur der digitalen Echtzeit-Animationen." Accessed March 22, 2021. https://www.unesco.de/kultur-und-natur/immaterielles-kulturerbe/immaterielles-kulturerbe-deutschland/demoszene.

"Dexion X-Mas Conference 90." C-64 Scene Database, 1990. https://csdb.dk/event/?id=83.

Dietz, Tracey L. "An Examination of Violence and Gender Role Portrayals in Video Games: Implications for Gender Socialization and Aggressive Behavior." *Sex Roles: A Journal of Research* 38, no. 5–6 (1998): 425–442.

Dijkstra, Edsger Wybe. "Go To Statement Considered Harmful." *Communications of the ACM*, March 1968.

Doherty, Neil F., Crispin R. Coombs, and John Loan-Clarke. "A Re-Conceptualization of the Interpretive Flexibility of Information Technologies: Redressing the Balance between the Social and the Technical." *European Journal of Information Systems* 15, no. 6 (2006): 569–582.

"Dominators/Upfront/Trilogy Party 1989." C-64 Scene Database, 1988. https://csdb.dk/event/?id=222.

"Dominators/Upfront/Trilogy Party 1989 Results." C-64 Scene Database, 1989. https://csdb.dk/release/?id=121620.

Donovan, Tristan. *Replay: The History of Video Games*. Lewes, UK: Yellow Ant, 2010.

Driscoll, Kevin. *The Modem World: A Prehistory of Social Media*. New Haven, CT: Yale University Press, 2022.

Driscoll, Kevin, and Joshua Diaz. "Endless Loop: A Brief History of Chiptunes." *Transformative Works and Cultures* 2 (2009).

Džin. "ARMSID," 2023. Nobomi. https://www.nobomi.cz/8bit/armsid/index_en.php.

Elko, Lance. "Editor's Notes." *Compute!'s Gazette*, May 1990.

"Eng und holzig." Der Spiegel, January 25, 1987, sec. Wirtschaft. https://www.spiegel.de/wirtschaft/eng-und-holzig-a-3d227296-0002-0001-0000-000013522457.

Engelbart, Doug. "The Strategic Pursuit of Collective IQ." Presented at the Fall Joint Computer Conference, San Francisco, CA, December 9, 1968. http://archive.org/details/XD1941_1_95VannevarBushSymTape2_DougEngelbart.

Fassonne, Riccardo. "Cammelli and Attack of the Mutant Camels." *Well Played* 6, no. 2 (June 25, 2017): 55–71.

"Fatal News #04." C-64 Scene Database, 1990. https://csdb.dk/release/?id=5620.

"The February Charts!" *ZZap!64*, February 1992.

Fuller, Matthew. *Media Ecologies: Materialist Energies in Art and Technoculture*. Cambridge, MA: MIT Press, 2007.

Garda, Maria B., and Paweł Grabarczyk. "But Our Princess Is on Another Platform . . ." In *Proceedings of DIGRA 2022: The 14th Digital Games Research Association Conference*. Kraków, Poland, 2022.

Gates, Bill. “An Open Letter to Hobbyists.” *Homebrew Computer Club Newsletter*, January 31, 1976.

“Bill Gates Interview.” National Museum of American History, Smithsonian Institution, 1993. https://americanhistory.si.edu/comphist/gates.htm.

Gazzard, Alison. *Now the Chips Are Down: The BBC Micro*. Cambridge, MA: MIT Press, 2016.

———. “Remembering Repton: An Alternative History of Co-Creativity in 1980s Britain.” *Kinephanos*, June 2015. http://www.kinephanos.ca/2015/remembering-repton-an-alternative-history-of-co-creativity-in-1980s-britain/.

Gehrmann, Holger. “Hyperscreen.” *64’er Magazin*, December 1985.

Gerli, Damiano. “A Discussion on the European Gaming Market in the 80s.” *Genesis Temple* (blog), April 30, 2022. https://genesistemple.com/playing-it-the-european-way-a-discussion-on-the-european-gaming-market-in-the-80s.

———. “A Swashbuckling Tale of Italian Software Piracy—1983–1993.” *Genesis Temple* (blog), June 11, 2020. https://genesistemple.com/a-swashbuckling-tale-of-italian-software-piracy-1983-1993.

Gingold, Chaim. *Building SimCity: How to Put the World in a Machine*. Cambridge, MA: MIT Press, 2024.

“Graphics Modes.” C64 Wiki. Accessed April 27, 2023. https://www.c64-wiki.com/wiki/Graphics_Modes.

Grip, Johan. “Introducing the C128 Neo.” *C128 Neo* (blog), November 27, 2019. https://c128.se/posts/introducing-the-c128-neo/.

Haddon, Leslie. “Electronic and Computer Games, the History of an Interactive Medium.” *Screen* 29, no. 2 (1988): 52–75.

Haddon, Leslie, and David Skinner. “The Enigma of the Micro: Lessons from the British Home Computer Boom.” *Social Science Computer Review* 9, no. 3 (October 1, 1991): 435–449.

Haigh, Thomas. “Becoming Universal.” *Communications of the ACM*, February 2022.

———. “The History of Information Technology.” *Annual Review of Information Science and Technology* 45, no. 1 (2011): 431–487.

Haigh, Thomas, and Paul E. Ceruzzi. *A New History of Modern Computing*. Cambridge, MA: MIT Press, 2021.

Halfhill, Tom R. “A Turning Point for Atari.” *Compute!*, April 1986.

———. “RIP Commodore. 1954–1994. A Look at an Innovative Industry Pioneer, Whose Achievements Have Been Largely Forgotten.” *BYTE*, August 1994.

———. “The MS-DOS Invasion: IBM Compatibles Are Coming Home.” *Compute!*, December 1986.

———. “Word Processing in The Home.” *Compute!’s Gazette*, January 1984.

Hammond, Steve. “GTA Origins.” *Stevehammond.Org* (blog), January 4, 2021. https://www.stevehammond.org/dmadesign/2021/01/04/Amiga-Addict-Issue2.html.

Hardisty, Mark James. *A Gremlin in the Works: 1983–1989*. Edited by Mark Green. Bath, UK: Bitmap Books, 2015.

Hartmann, Doreen. “Animation in the Demoscene: From Obfuscation to Category (Or: How to Demonstrate Skills without Adhering to the Real-Time Principle).” *WiderScreen* 17, no. 1–2 (2014). http://widerscreen.fi/numerot/2014-1-2/animation-demoscene-obfuscation-category-demonstrate-skills-without-adhering-real-time-principle/.

———. *Digital Art Natives: Praktiken, Artefakte und Strukturen der Computer-Demoszene*. Berlin, Germany: Kulturverlag Kadmos Berlin, 2017.

Hartung, Matthias. “Szene Inside.” *64’er Magazin*, March 1995.

Hastik, Canan, and Arnd Steinmetz. “Demoscene Computer Artists and Community.” In *Proceedings of CERC2012 Collaborative European Research Conference*, ed. Patrick Bours, Bernard Humm, Robert Loew, Ingo Stengel, and Paul Walsh. Darmstadt, Germany, 2012.

hedning. “Gubbdata 10 Years.” *Propaganda*, July 27, 2022.

Hege, Nordli. “The Gathering–Computer Parties as Means for Gender Inclusion.” IST-2000–26329 SIGIS. European Commission, 2003.

Henriksen, Lars Skov, Inger Koch-Nielsen, and David Rosdahl. “Formal and Informal Volunteering in a Nordic Context: The Case of Denmark.” *Journal of Civil Society* 4, no. 3 (December 1, 2008): 193–209.

Herd, Bil, Albert Charpentier, and Stephen Edwards. “Commodore VIC I and VIC II Chips.” Presented at the Vintage Computer Festival East, Wall Township, NJ, April 22, 2022. https://www.youtube.com/watch?v=QNLbi6ZanUE.

Herd, Bil, and Margaret Gorts Morabito. *Back into the Storm: A Design Engineer’s Story of Commodore Computers in the 1980s*. Independently published, 2021.

Hiltzik, Michael A. *Dealers of Lightning: Xerox PARC and the Dawn of the Computer Age*. New York: Harper Business, 1999.

“Hitline in Februar.” *Aktueller Software Markt (ASM)*, February 1992.

Home & Personal Computers of America. “Final Cartridge III: All New, Now with Windows!” *Run*, November 1987.

“HOTLINE: Top 30 Im Mai.” *Aktueller Software Markt (ASM)*, April 1986.

“HOTWARE: Software Best Sellers.” *Compute!*, November 1985.

“HOTWARE: Software Best Sellers.” *Compute!*, September 1986.

“HOTWARE: Software Best Sellers.” *Compute!*, December 1986.

Hubbard, Rob. “Golden Days of Computer Game Music (Part 1).” September 5, 2007. https://www.youtube.com/watch?v=DiPdjbsiQqM.

———. “Rob Hubbard.” C64.COM, 2007. http://www.c64.com/interviews/hubbard_part_2.html.

———. “Interview by 6581SID (KDVS).” Internet Archive WayBack Machine, September 28, 1999. https://web.archive.org/web/20060210115105/http://www.freenetpages.co.uk/hp/tcworh/int_6581.htm.

Hülsbeck, Chris. “Musik . . . Wie Noch Nie.” *64’er Magazin*, October 1986.

Intangible Cultural Heritage. “Demoscene, Musical Saw Playing and Horsemanship of the Roma—12 New Elements Inscribed on the National Inventory of Living Heritage.” Museovirasto, April 9, 2020. https://www.museovirasto.fi/en/articles/demoskene-sahansoitto-ja-romanien-hevostaidot-elavan-perinnon-kansalliseen-luetteloon-12-uutta-kohdetta.

“Interview: Peter Liepa.” Boulder Dash, 2005. http://www.boulder-dash.nl/interview.php?lang=en.

“Interview with Joost Honig (Honey).” C64.COM, 2004. https://web.archive.org/web/20040608093152fw_/http://www.c64hq.com/interviews/honig.html.

“Investment Package.” *Personal Computer Games*, Summer 1983, 8–12.

Isaacson, Walter. *The Innovators: How a Group of Hackers, Geniuses, and Geeks Created the Digital Revolution*. New York: Simon & Schuster, 2014.

Jenkins, David. “Gremlin Graphics Presents: Bounder’s World and the Return of Super Cars.” *Metro* (blog), July 22, 2011. https://metro.co.uk/2011/07/22/gremlin-graphics-presents-bounders-world-and-the-return-of-super-cars-88959/.

Jones, Darran, ed. *The Commodore 64 Book: 30th Anniversary Special*. Bournemouth, UK: Imagine Publishing, 2012.

Jørgensen, Kristine, Ulf Sandqvist, and Olli Sotamaa. “From Hobbyists to Entrepreneurs: On the Formation of the Nordic Game Industry.” *Convergence* 23, no. 5 (October 1, 2017): 457–476.

Juul, Jesper. *Half-Real: Video Games between Real Rules and Fictional Worlds*. Cambridge, MA: MIT Press, 2005.

———. *Handmade Pixels: Independent Video Games and the Quest for Authenticity*. Cambridge, MA: MIT Press, 2019.

———. “The Game, the Player, the World: Looking for a Heart of Gameness.” In *Level Up: Digital Games Research Conference Proceedings*, edited by Marinka Copier and Joost Raessens, 30–45. Utrecht, the Netherlands: Utrecht University, 2003. http://www.jesperjuul.net/text/gameplayerworld/.

———. “Without a Goal: On Open and Expressive Games.” In *Videogame, Player, Text*, ed. Barry Atkins and Tanya Krzywinska, 191–203. Manchester, UK: Manchester University Press, 2007. http://www.jesperjuul.net/text/withoutagoal.

Juul, Jesper, and Laurel Carney. “Would You Like Games with That Computer? Revisiting Early Game History & Culture with the Commodore 64.” In *Proceedings of the DiGRA 2023 Conference*. Seville, Spain, 2023. https://www.jesperjuul.net/text/gameswithc64/.

Juul, Jesper, and Marleigh Norton. “Easy to Use and Incredibly Difficult: On the Mythical Border between Interface and Gameplay.” In *Proceedings of the 4th International Conference on Foundations of Digital Games*, 107–112. Orlando, FL: ACM, 2009. http://portal.acm.org/citation.cfm?id=1536539.

Keizer, Gregg, ed. *Compute!’s Machine Language Games for the Commodore 64*. Greensboro, NC: Compute! Publications, 1986.

Kelly, J. “Impossible Scroll.” *Compute!’s Gazette*, September 1987.

Kemeny, John G. *Man and the Computer*. New York: Scribner, 1972.

Kenniscentrum Immaterieel. “215e bijschrijving in Inventaris Immaterieel Erfgoed Nederland.” *Immaterieel Erfgoed* (blog), July 3, 2023. http://www.immaterieelerfgoed.nl/nl/215e-bijschrijving-in-inventaris.

Kent, Steven L. *The Ultimate History of Video Games: From Pong to Pokemon and beyond . . . the Story behind the Craze That Touched Our Lives and Changed the World*. New York: Three Rivers Press, 2010.

Kidder, Tracy. *The Soul of a New Machine*. Boston, MA: Little, Brown, 1981.

Kim Justice. “Commodore 64 vs ZX Spectrum—The Great British Computer War.” February 14, 2016. https://www.youtube.com/watch?v=wUnjoztg1Fg.

Kirby, David. “The Future Is Now: Diegetic Prototypes and the Role of Popular Films in Generating Real-World Technological Development.” *Social Studies of Science* 40, no. 1 (February 1, 2010): 41–70.

Kirchesch, Christian. “Chris Hülsbeck Über Turrican, Great Giana Sisters.” *Digitalista* (blog), February 11, 2015. https://web.archive.org/web/20181001142258/https://digitalista.de/interviews/2015/Februar/11/chris-hulsbeck-uber-turrican-great-giana-sisters-1.html.

Kirkpatrick, Graeme. “Constitutive Tensions of Gaming’s Field: UK Gaming Magazines and the Formation of Gaming Culture 1981–1995.” *Game Studies* 12, no. 1 (2012). http://gamestudies.org/1201/articles/kirkpatrick.

———. *The Formation of Gaming Culture*. Basingstoke, UK: Palgrave Pivot, 2015.

Klein, Peter. “Assembler Corner: Zeichensatz & Scrolling.” *64’er Magazin*, October 1992.

———. “Profi-Corner.” *64’er Magazin*, May 1992.

Kluitenberg, Eric. “On the Archaeology of Imaginary Media.” In *Media Archaeology Approaches, Applications, and Implications*, ed. Erkki Huhtamo and Jussi Parikka, 48–69. Berkeley, CA: University of California Press, 2011.

Kubrick, Stanley, dir. *2001: A Space Odyssey*. Metro-Goldwyn-Mayer (MGM), Stanley Kubrick Productions, 1968.

Kuorikoski, Juho. *Commodore 64: Tasavallan tietokone*. Helsinki, Finland: Minerva Kustannus, 2017.

Lancaster, Don. “TV Typewriter.” *Radio-Electronics*, September 1973.

“Lemmings: An EXCLUSIVE Demo from Psygnosis!” *Commodore Force*, January 1993.

Levy, Steven. *Hackers: Heroes of the Computer Revolution*. Garden City, NY: Anchor Press Doubleday, 1984.

Lft. “Safe VSP.” C-64 Scene Database, 2013. https://csdb.dk/release/?id=115690.

Lichnerowicz, Andrzej. “Komitet Społeczny Kronika Polskiej Demosceny.” Komitet Społeczny Kronika Polskiej Demosceny, December 15, 2021. https://kskpd.pl/.

Liddon, Gary. “The Genius of Amplified Light.” *Zzap!64*, July 1986.

———. “Transporting Sprites Across the Border.” *Zzap!64*, August 1987.

Light, Jennifer S. “When Computers Were Women.” *Technology and Culture* 40, no. 3 (1999): 455–483.

Lock, Robert C., and Tom R. Halfhill. “Editor’s Notes.” *Compute!*, July 1986.

Lyons, Paul. “A Short History of the Demo . . .” *Commodore Format*, November 1990.

Maclean, Archer, and Ant & Nic’s Cabin Fever. “Creating International Karate & IK+.” September 9, 2022. https://www.youtube.com/watch?v=zRFZS9wzt7s.

“Madonna Cracking Group Party.” C-64 Scene Database, 1988. https://csdb.dk/event/?id=730.

Magaudda, Paolo, and Sergio Minniti. “Retromedia-in-Practice: A Practice Theory Approach for Rethinking Old and New Media Technologies.” *Convergence* 25, no. 4 (August 1, 2019): 673–693.

Maher, Jimmy. *The Future Was Here: The Commodore Amiga*. Cambridge, MA: MIT Press, 2012.

———. “Will Wright’s City in a Box.” *The Digital Antiquarian* (blog), June 10, 2016. https://www.filfre.net/2016/06/simcity-part-1-will-wrights-city-in-a-box/.

Mailland, Julien, and Kevin Driscoll. *Minitel: Welcome to the Internet*. Cambridge, MA: MIT Press, 2017.

“Manic Miner Review.” *Zzap!64*, June 1989.

Mansoux, Aymeric, Brendan Howell, Dušan Barok, and Ville-Matias Heikkilä. “Permacomputing Aesthetics: Potential and Limits of Constraints in Computational Art, Design and Culture.” In *Computing within Limits*, 2023. https://limits.pubpub.org/pub/6loh1eqi/release/1.

Mast, Patrick. “Final Cartridge III Manual.” Riska B. V. Home & Personal Computers, 1987.

“Matthew Smith Interview Manic Miner ZX Spectrum.” 2007. http://www.youtube.com/watch?v=FWmmMZlhcqU.

McMurtrie, Dave. “Building Wizard of Wor for the Commodore 64 from the Original 1983 Source Code.” May 23, 2023. https://www.youtube.com/watch?v=d7WO2kJ5ye0.

———. “Gail Wellington: Far More than Just a Herder of CATS and Mother of CDTV.” *Commodore International Historical Society* (blog), December 12, 2022. https://commodore.international/2021/11/21/gail-wellington-far-more-than-just-a-herder-of-cats-and-mother-of-cdtv/.

———. “The Commodore Clipper (and Other Tales of the Summer 1983 CES).” *Commodore International Historical Society* (blog), December 5, 2021. https://commodore.international/2021/08/22/the-commodore-clipper-and-other-tales-of-the-summer-1983-ces/.

McSweeney, Anthony. “Rob Hubbard’s Music: Disassembled, Commented and Explained.” *C= Hacking*, March 7, 1993. https://codebase64.org/doku.php?id=magazines:chacking5#rob_hubbard_s_musicdisassembled_commented_and_explained.

Mensch, Bill. “In Memoriam of Charles ‘Chuck’ Peddle.” *The Western Design Center, Inc*. (blog), December 20, 2019. https://wdc65xx.com/in-memoriam-of-chuck-peddle/.

Megler, Veronika. “There and Back Again: A Case History of Writing ‘The Hobbit.’” Paper presented at *Born Digital and Cultural Heritage Conference*, Melbourne, Victoria, Australia, 2014. http://veronikamegler.com/BDCH_2014_Megler_The_Hobbit_v1.1.pdf.

Meyers, Raquel. “Keys of Fury—Type in Beyond the Scrolling Horizon.” *WiderScreen* 20, no. 1–2, 2017. http://widerscreen.fi/numerot/2017-1-2/keys-of-fury-type-in-beyond-the-scrolling-horizon/.

Minter, Jeff. “Psychedelia—History,” 2005. http://www.minotaurproject.co.uk/psychedelia.php.

Minter, Jeff, and Andrew Fisher. “Foreword.” In *The Commodore 64 Book—1982 to 199x*, ed. Andrew Rollings, i. Auburn, AL: hiive, 2007.

MobyGames. “About.” Accessed January 24, 2024. https://www.mobygames.com/info/about/.

Moleman. “Stamps Back.” December 30, 2022. https://www.youtube.com/watch?v=YUqn1OPxtmE.

Monopolies and Mergers Commission. “Market Size and Market Shares. Video Games. A Report on the Supply of Video Games in the UK,” 1995.

Montfort, Nick. *Twisty Little Passages: An Approach to Interactive Fiction*. Cambridge, MA: MIT Press, 2003.

Montfort, Nick, Patsy Baudoin, John Bell, Ian Bogost, Jeremy Douglass, Mark C. Marino, Michael Mateas, Casey Reas, Mark Sample, and Noah Vawter. *10 PRINT CHR$(205.5+RND(1)); : GOTO 10*. Cambridge, MA: MIT Press, 2012.

Montfort, Nick, and Ian Bogost. *Racing the Beam: The Atari Video Computer System*. Cambridge, MA: MIT Press, 2009.

“Monty on the Run Review.” *Zzap!64*, October 1985.

Morabito, Margaret. “Enter the Online World of LucasFilm.” *Run*, August 1986.

Morningstar, Chip, and F. Randall Farmer. “The Lessons of Lucasfilm’s Habitat.” *Journal For Virtual Worlds Research* 1, no. 1 (1991–2008). https://jvwr-ojs-utexas.tdl.org/jvwr/article/view/287.

Morrison, Paul. “8-Bit Symphony and Rob Hubbard.” September 11, 2019. https://www.youtube.com/watch?v=lNaU2F57ptE.

“Most Important Companies.” *BYTE*, September 1995.

Mr. Cursor. “Reactions to the MCG Party.” *Illegal*, May 1988.

mrsid. “Prince of Persia for Commodore 64/128 Released.” *Prince of Persia C64—Development Blog* (blog), October 16, 2011. http://popc64.blogspot.com/2011/10/prince-of-persia-for-

commodore-64128.html.

M. W. “Boulder Dash CBM 64 £8.95.” *Home Computing Weekly*, October 1984.

Natale, Simone, and Gabriele Balbi. “Media and the Imaginary in History.” *Media History* 20, no. 2 (April 3, 2014): 203–218.

Navarro-Remesal, Víctor, and Óliver Pérez-Latorre, eds. *Perspectives on the European Videogame*. Amsterdam: Amsterdam University Press, 2021.

neogames. “The Game Industry of Finland Report 2020,” 2021. https://neogames.fi/wp-content/uploads/2021/06/FGIR-2020-Report.pdf.

“Neue Tauschpartner!!” *Illegal #19*, 1987.

Newman, James. “Cuthbert Goes Cloning: Ports, Platforms, and the Dragon 32 Microcomputer.” In *Perspectives on the European Videogame*, ed. Víctor Navarro-Remesal and Óliver Pérez-Latorre, 111–128. Amsterdam, the Netherlands: Amsterdam University Press, 2021.

Ninja / The Dreams. “Behind the Scenes of a C64 Demo.” Presented at the 28c3, 2011.

Nintendo. “Nintendo Co., Ltd. Consolidated Sales Transition by Region,” 2012. http://www.nintendo.co.jp/ir/library/historical_data/pdf/consolidated_sales_e1209.pdf.

Nishi, Kazuhiko. “[Legendary PC MSX] The Creator Finally Reveals the ‘Essence of Failure’” *Diamond Online* (blog), December 26, 2020. https://diamond.jp/articles/-/253320.

Noel, Lee, Jr. “GEOS: The Quiet Revolution.” *Compute!’s Gazette*, June 1986.

Nooney, Laine. *The Apple II Age: How the Computer Became Personal*. Chicago, IL: University of Chicago Press, 2023.

“Ny Lag Stoppar Pirat-Kopiering.” *Datormagasin*, May 1986.

Oliveira, Bruno, Artur Franco, Wellington Franco, Fernando Carvalho-Gomes, José Maia, and Gilvan Ferreira. “A Framework for Metroidvania Games.” In *Proceedings of SBGames 2020*, 836–844. Recife, Brazil, 2020.

Parikka, Jussi. *What Is Media Archaeology*. Cambridge, UK: Polity, 2013.

“The Party 2002.” C-64 Scene Database, 2002. https://csdb.dk/event/?id=514.

Peddle, Chuck. “Lecture and Presentation of the Victor 9000 Computer.” Presented at the Boston Computer Society, New England Life Hall, October 27, 1982. https://www.youtube.com/watch?v=i_z443gwtW0.

———. “The 6502 to the PC.” Presented at the Vintage Computer Festival Southeast, April 30, 2017. https://www.youtube.com/watch?v=zUn3isBLkTE.

“Penguins Make Life Perilous!!—Manic Miner.” *Computers & Video Games*, September 1983.

Perry, Tekla S., and Paul Wallich. “Design Case History: The Commodore 64.” *IEEE Spectrum* 22, no. 3 (March 1985): 48–58.

Pleasance, David. *Commodore The Inside Story: The Untold Tale of a Computer Giant*. Lewes, UK: Unicorn, 2021.

Pleasance, David John, and Trevor Dickinson. *From Vultures to Vampires: 25 Years of Copyright Chaos and Technology Triumphs, Volume One: 1995–2004*. Lewes, UK: Unicorn, 2022.

Polgár, Tamár. *Freax: The Brief History of the Computer Demoscene*. Winnenden, Germany: CSW-Verlag, 2005.

Pollack, Andrew. “The Home Computer Arrives.” *New York Times*, June 17, 1982.

Pouët.net, ”pouët.net 2.0 API.” Accessed March 15, 2024. https://api.pouet.net/.

Price, James. “Editorial.” *Commodore Force*, March 1994.

RadioShack. “A Few Years Ago, the Idea of a Computer You Could Put in Your Pocket Was Just Science Fiction.” 1980.

Rahm-Skågeby, Jörgen, and Anders Carlsson. “The Archive and the Scene: On the Cultural Techniques of Retrocomputing Databases.” *New Media & Society* 23, no. 4 (April 1, 2021): 732–749.

Randall, Neil. “The Last Crusade.” *Compute!*, December 1989.

Rankin, Joy Lisi. *A People’s History of Computing in the United States*. Cambridge, MA: Harvard University Press, 2018.

“Raubkopieren.” *Highscore*. WDR, December 17, 1988. https://www.youtube.com/watch?v=m3pscDZOIAM.

Raymond, Eric S. *The New Hacker’s Dictionary*. Cambridge, MA: MIT Press, 1996.

Reggio, Godfrey, dir. *Koyaanisqatsi*. American Zoetrope, IRE Productions, Santa Fe Institute for Regional Education, 1982.

Retro Recipes. “All New Commodore 64 Using Ultimate 64 & LEGO Keycaps!” March 9, 2019. https://www.youtube.com/watch?v=5UX-gqylYgQ.

Retrospective. “EVO64 is a completely modern, yet still fully authentic Commodore 64 remake, with tons of new capabilities.” Accessed January 19, 2024. https://retrospective.shop/evo64-rev1/.

Reunanen, Markku. “Computer Demos—What Makes Them Tick?” Licentiate, Aalto University, 2010.

———. “How Those Crackers Became Us Demosceners.” *WiderScreen* 17, no. 2 (April 15, 2014).

———. “Studying Yourself—Case Demoscene.” *Centre of Excellence in Game Culture Studies* (blog), September 8, 2023. https://coe-gamecult.org/2023/09/08/markku-reunanen-studying-yourself-case-demoscene/.

———. “Times of Change in the Demoscene: A Creative Community and Its Relationship with Technology.” PhD dissertation, University of Turku, 2017.

Reunanen, Markku, Tero Heikkinen, and Anders Carlsson. “PETSCII—A Character Set and a Creative Platform.” *Replay* 5, no. 1 (2018): 27–47.

Reunanen, Markku, and Antti Silvast. “Demoscene Platforms: A Case Study on the Adoption of Home Computers.” *IFIP Advances in Information and Communication Technology*, 2007, 289–301.

Rignall, Julian. “The Compunet Page.” *Zzap!64*, December 1986.

Rossi, Randy. “VIC-II Kawari.” Accentual, 2022. http://accentual.com/vicii-kawari/.

Rothstein, Bo. “Corruption and Social Trust: Why the Fish Rots from the Head Down.” *Social Research* 80, no. 4 (2013): 1009–1032.

Saarikoski, P., and J. Suominen. “Computer Hobbyists and the Gaming Industry in Finland.” *IEEE Annals of the History of Computing* 31, no. 3 (2009): 20–33.

Samar Crew. “$11 Music Compo.” C64 Scene Database, 2018. https://csdb.dk/forums/?roomid=12&topicid=127905.

Schramm, Karsten, and Boris Schneider. “Der Neue.” *64’er Magazin*, June 1986.

“Segas Sell Better than Nintendos—Official!” *Sega Power*, May 1991.

Silicon Limited, Success + The Ruling Company, and Xenon. “X’2023.” Commodore 64 Scene Database, June 2, 2023. https://csdb.dk/event/?id=3187.

Silveria, Terry. “Spreadsheets for the C-64.” *Ahoy!*, May 1984.

“SimCity Review.” *Zzap!64*, February 1990.

Skågeby, Jörgen, and Lina Rahm. “What Is Feminist Media Archaeology?” *Communication +1* 7, no. 1 (October 4, 2018). https://scholarworks.umass.edu/cpo/vol7/iss1/7.

SNC. “Son of Blagger Review.” *Personal Computer Games*, July 1984.

"Software Reviews: Impossible Mission." *Computer & Video Games*, April 1985.

Sørensen, Lars Roger. "Udmattede Unge EDB-Entusiaster." *Jylland-Posten*, December 30, 1990.

Stachniak, Zbigniew. "Red Clones: The Soviet Computer Hobby Movement of the 1980s." *IEEE Annals of the History of Computing* 37, no. 1 (January 2015): 12–23.

Steil, Michael. "How Many Commodore 64 Computers Were Really Sold?" *Pagetable.Com* (blog), February 1, 2011. https://www.pagetable.com/?p=547.

Stein, Jesse Adams. "Domesticity, Gender and the 1977 Apple II Personal Computer." *Design and Culture* 3, no. 2 (2011): 193–216.

Street Tuff. "ECM Mode Demoeffect Competition." C64 Scene Database, 2020. https://csdb.dk/event/?id=2998.

Success. "Lethal News #1." C64 Scene Database, 1990. https://csdb.dk/release/?id=8755.

Sumner, James. "What Makes a PC? Thoughts on Computing Platforms, Standards, and Compatibility." *IEEE Annals of the History of Computing* 29, no. 2 (April 2007): 88.

Švelch, Jaroslav. "Keeping the Spectrum Alive." In *Fans and Videogames: Histories, Fandom, Archives*, ed. Melanie Swalwell, Helen Stuckey, and Angela Ndalianis, 57–74. New York: Routledge, 2017.

Swalwell, Melanie. *Homebrew Gaming and the Beginnings of Vernacular Digitality*. Game Histories. Cambridge, MA: MIT Press, 2021.

"Talking with Manfred Trenz." retrogaming.it, March 17, 2004. https://archive.ph/zkcps.

Tasajärvi, Lassi. *Demoscene: The Art of Real-Time*. Helsinki, Finland: Even Lake Studios, 2004.

Taylor, T. L., and Emma Witkowski. "This Is How We Play It: What a Mega-LAN Can Teach Us About Games." Foundations of Digital Games Conference, 2010.

The Pirates Crew. "Pirates #08." C64 Scene Database, 1990. https://csdb.dk/release/?id=79955.

Sen, Paul, dir. *The Triumph of the Nerds: The Rise of Accidental Empires*. John Gau Productions, Channel 4, PBS, 1996.

Therrien, Carl. "Video Games Caught Up in History." In *Before the Crash: Early Video Game History*, ed. Mark J. P. Wolf, 9–29. Detroit, MI: Wayne State University Press, 2012.

Thissen, Torsten. "Data Becker—Das Ende Einer Institution." *RP Online* (blog), October 11, 2013. https://web.archive.org/web/20131016091132/http://www.rp-online.de/region-duesseldorf/duesseldorf/nachrichten/data-becker-das-ende-einer-institution-1.3738335.

Tomczyk, Michael S. "Preliminary Recommendations on the VIC Product Line," June 20, 1980. https://archive.org/details/vic-preliminary-recommendations/.

———. *The Home Computer Wars: An Insider's Account of Commodore and Jack Tramiel*. Greensboro, NC: Compute, 1984.

Tommervik, Allan. "The Great Arcade / Computer Controversy. Part 1: The Publishers and the Pirates." *Softline*, January 1982.

"Top Twenty." *Popular Computing Weekly*, November 7, 1985.

Tramiel, Leonard. "An Inexcusable Failure." *Vintage Computer Stories* (blog), September 4, 2022. https://vintagecomputerstories.blogspot.com/2022/09/an-inexcusable-failure.html.

———. "Creating PETSCII." *Vintage Computer Stories* (blog), December 27, 2021. https://vintagecomputerstories.blogspot.com/2021/12/creating-petscii.html.

Tramiel, Leonard, and Dave McMurtrie. "Early Commodore History." Presented at the Vintage Computer Festival West, Mountain View, CA, August 5, 2023. http://archive.org/details/VCFW2023_Early_Commodore_History_-_Leonard_Tramiel_and_Dave_McMurtrie.

Tyni, Heikki, and Olli Sotamaa. “Assembling a Game Development Scene? Uncovering Finland’s Largest Demo Party.” *G|A|M|E Games as Art, Media, Entertainment* 1, no. 3 (2014).

Tyschtschenko, Petro, and Patric Kloeter. *Meine Erinnerungen an Commodore und Amiga*. Rödermark, Germany: Power Service, 2014.

Uricchio, William. “Television’s First Seventy-Five Years: The Interpretive Flexibility of a Medium in Transition.” In *The Oxford Handbook of Film and Media Studies*, ed. Robert Kolker, 286–305. Oxford, UK: Oxford University Press, 2008.

Varga, Andreas. “Interview with Bob Yannes,” August 30, 1996. https://web.archive.org/web/20060210052807/http://stud4.tuwien.ac.at/~e9426444/yannes.html.

Vintage Computer Federation. “Commodore Employees Panel.” Vintage Computer Festival East, July 29, 2022. https://www.youtube.com/watch?v=8cvu_iRvM-Y.

Vose, Michael G. “Software Review: True BASIC.” *BYTE*, May 1985.

Wade, Alex. *Playback—A Genealogy of 1980s British Videogames*. London: Bloomsbury, 2016.

Wasiak, Patryk. “‘Amis and Euros.’ Software Import and Contacts Between European and American Cracking Scenes.” *WiderScreen* 17, no. 1–2 (2014). http://widerscreen.fi/numerot/2014-1-2/amis-euros-software-import-contacts-european-american-cracking-scenes/.

———. “Computer Dealer Demos: Selling Home Computers with Bouncing Balls and Animated Logos.” *IEEE Annals of the History of Computing* 35, no. 4 (October 2013): 56–68.

———. “‘Illegal Guys’: A History of Digital Subcultures in Europe during the 1980s.” *Zeithistorische Forschungen*, 2012. https://zeitgeschichte-digital.de/doks/1595.

Webster, Bruce. “Microcomputer Color Graphics—Observations.” *BYTE*, December 1985.

“Wie eine Epidemie.” Der Spiegel, April 17, 1983, sec. Wirtschaft. https://www.spiegel.de/wirtschaft/wie-eine-epidemie-a-8011539f-0002-0001-0000-000014019839.

Wierzbicki, Barbara. “Longevity of Commodore 64, VIC 20 Questioned.” *InfoWorld*, December 5, 1983.

Wilkins, Chris. *The History of Ocean Software*. Kenilworth, UK: Fusion Retro Books, 2013.

Wilkins, Chris, and Roger M. Kean. *Let’s Go Dizzy: The Story of the Oliver Twins*. Kenilworth, UK: Fusion Retro Books, 2016.

———. *The Story of US Gold*. Kenilworth, UK: Fusion Retro Books, 2015.

Williams, Ken. *Not All Fairy Tales Have Happy Endings: The Rise and Fall of Sierra On-Line*. Independently published, 2020.

Wolf, Mark J. P., ed. *Before the Crash: Early Video Game History*. Detroit, MI: Wayne State University Press, 2012.

Wozniak, Steve, and Gina Smith. *iWoz: Computer Geek to Cult Icon: How I Invented the Personal Computer, Co-Founded Apple, and Had Fun Doing It*. New York: W. W. Norton, 2006.

Wszola, Stan. “Hardware Review: Commodore 64.” *BYTE*, July 1983.

WWS/RWE + J. S. “‘Illegal’ Presents the OLD 64 LEGENDS! This Month: Cracking 1982–1985.” *Illegal #30*, 1988.

ZZAP69. “Gubbdata 2022.” *Propaganda*, 2022.

Zenit Artivisive, Docabout, “Programma 101- Memory of Future,” 2010. https://www.youtube.com/watch?v=lpkqdbz1R_s.

Index

Your gateway to knowledge and culture. Accessible for everyone.