

Racing the Beam

The Atari Video Computer System

Nick Montfort and Ian Bogost

The MIT Press Cambridge, Massachusetts London, England

© 2009 Nick Montfort and Ian Bogost

All rights reserved. No part of this book may be reproduced in any form by any electronic or mechanical means (including photocopying, recording, or information storage and retrieval) without permission in writing from the publisher.

For information about special quantity discounts, please email special_sales@mitpress.mit.edu

This book was set in Filosofia and Helvetica Neue by SNP Best-set Typesetter Ltd., Hong Kong.

Printed and bound in the United States of America.

Library of Congress Cataloging-in-Publication Data

Montfort, Nick.

Racing the beam : the Atari video computer system / Nick Montfort and Ian Bogost.

p. cm — (Platform studies)

Includes bibliographical references and index.

ISBN 978-0-262-01257-7 (hardcover : alk. paper) 1. Video games—Equipment and supplies.
2. Atari 2600 (Video game console) 3. Computer games—Programming. 4. Video games—
United States—History. I. Bogost, Ian. II. Title.

TK6681.M65 2009

794.8—dc22

2008029410

10 9 8 7 6 5 4 3 2 1

When someone creates a computer artifact like a video game, a digital artwork, or a work of electronic literature, what type of process is this? Here's one idea: it is a creative act that is similar in many ways to writing a poem or taking a photograph, except that in this case, the creator doesn't use words one after another on paper or light bent through an aperture. This type of inscription or exposure doesn't happen—so what exactly does happen?

The creator of a computer work might design circuits and solder chips. Or, this author might write instructions for the integrated circuits and microprocessors of a particular computer, or write software in a high-level programming language, or create 3D models to be added to a virtual world, or edit digital video for embedding in a Web site.

The same question could be asked of the critic who interacts with such a work. What does a creator, historian, researcher, student, or other user do when experiencing a creative computer artifact? An encounter with such a work could involve trying to understand the social and cultural contexts in which it came to exist. It might also involve interpreting its representational qualities—what it means and how it produces that meaning. Alternatively, a study might involve looking at the methods of this work's construction, or the code itself, or even the hardware and physical form of the machines on which it is used.

All of these levels of computational creativity are connected. Fortunately for those of us who are interested in such uses of the computer, there have already been many studies of digital media dealing with the

reception and operation of computer programs, with their interfaces, and with their forms and functions. But studies have seldom delved into the code of these programs, and they have almost never investigated the platforms that are the basis of creative computing.¹ Serious and in-depth consideration of circuits, chips, peripherals, and how they are integrated and used is a largely unexplored territory for both critic and creator.

Platforms have been around for decades, though, right underneath our video games, digital art, electronic literature, and other forms of expressive computing. Digital media researchers are starting to see that code is a way to learn more about how computers are used in culture, but there have been few attempts to go even deeper, to investigate the basic hardware and software systems upon which programming takes place, the ones that are the foundation for computational expression. This book begins to do this—to develop a critical approach to computational platforms.

We hope this will be one of several considerations of this low level of digital media, part of a family of approaches called “platform studies.” Studies in this field will, we hope, investigate the relationships between platforms—the hardware and software design of standardized computing systems—and influential creative works that have been produced on those platforms.

Types of Platforms

The Atari Video Computer System (or VCS, a system also known by its product number, 2600) is a well-defined example of a platform. A platform in its purest form is an abstraction, a particular standard or specification before any particular implementation of it. To be used by people and to take part in our culture directly, a platform must take material form, as the Atari VCS certainly did. This can be done by means of the chips, boards, peripherals, controllers, and other components that make up the hardware of a physical computer system. The platforms that are most clearly encapsulated are those that are sold as a complete hardware system in a packaged form, ready to accept media such as cartridges. The Atari VCS is a very simple, elegant, and influential platform of this sort.

In other cases, a platform includes an operating system. It is often useful to think of a programming language or environment on top of an operating system as a platform, too. Whatever the programmer takes for granted when developing, and whatever, from another side, the user is required to have working in order to use particular software, is the plat-

form. In general, platforms are layered—from hardware through operating system and into other software layers—and they relate to modular components, such as optional controllers and cards. Studies in computer science and engineering have addressed the question of how platforms are best developed and what is best encapsulated in the platform. Studies in digital media have addressed the cultural relevance of particular software that runs on platforms. But little work has been done on how the hardware and software of platforms influences, facilitates, or constrains particular forms of computational expression.

When digital media creators choose a platform, they simplify development and delivery in many ways. For example, such authors need not construct an entirely new computer system before starting on a particular creative project. Likewise, users need not fashion or acquire completely new pieces of hardware before interacting with such a work. That said, work that is built for a platform is supported and constrained by what the chosen platform can do. Sometimes the influence is obvious: a monochrome platform can't display color, for instance, and a videogame console without a keyboard can't accept typed input. But there are more subtle ways that platforms influence creative production, due to the idioms of programming that a language supports or due to transistor-level decisions made in video and audio hardware. In addition to allowing certain developments and precluding others, platforms also function in more subtle ways to encourage and discourage different sorts of computer expression. In drawing raster graphics, there is a considerable difference between setting up one television scan line at a time as the Atari VCS demands, having a buffered display with support for tiles and sprites, or having some more elaborate system that includes a native 3D renderer. Such a difference can end up being much more important than simple statistics of screen resolution or color depth that are used as shorthand by fans and marketers.

We offer here such a platform study, one that considers an influential videogame system that helped introduce computing to a popular audience and to the home. Our approach is mainly informed by the history of material texts, programming, and computing systems. Other sorts of platform studies may emphasize different technical or cultural aspects, and may draw on different critical and theoretical approaches. To deal deeply with platforms and digital media, however, any study of this sort must be technically rigorous. The detailed analysis of hardware and code connects to the experience of developers who created software for a platform and users who interacted with and will interact with programs on that platform. Only the serious investigation of computing systems as

specific machines can reveal the relationships between these systems and creativity, design, expression, and culture.

Although it was not the first home videogame console, the Atari VCS was the first wildly popular one. It was affordable at the time, and it offered the flexibility of interchangeable cartridges. The popularity of the Atari VCS—which was the dominant system for years and remained widely used for more than a decade—supported the creation of nearly one thousand games, many of which established techniques, mechanics, or entire genres that continue to thrive today on much more technologically advanced platforms. Although several companies fielded consoles, by 1981 the Atari VCS accounted for 75 percent of home videogame system sales.² Indeed, the generic term for a videogame system in the early 1980s was “an Atari.” Yet, despite its undisputed place in the annals of popular culture, and despite having been the standard system for home video gaming for so many years, Atari’s first cartridge-based system is an extremely curious computer.

Cost concerns led to a remarkable hardware design, which influenced how software was written for the Atari VCS, which in turn influenced the video games created during and after the system’s reign. Given that it used a version of the very typical 6502 processor, which drove many computers and consoles, one might not guess that the Atari VCS was so atypical. But this processor interfaced with the display by means of a truly unique component, the Television Interface Adaptor, or TIA. A television picture is composed of many horizontal lines, illuminated by an electron beam that traces each one by moving across and down a picture tube. Some programmers worry about having each frame of the picture ready to be displayed on time; VCS programmers must make sure that each individual *line* of each frame is ready as the electron gun starts to light it up, “racing the beam” as it travels down the screen.

The Roots of Video Gaming

In *World of Warcraft*, you start off, as a human, in Northshire Abbey. You can move your character around using the W, A, S, and D keys, an interface popularized by the first-person shooter *Quake*. As you do this, the terrain that you’re standing on moves off the screen and new terrain appears as if from off screen. You are in a virtual space that is larger than the screen. This shouldn’t be at all surprising. It seems that every 3D game, from *Grand Theft Auto: San Andreas* and *Super Mario 64* to *Tomb Raider*, offers virtual spaces that are larger than the screen. *Quake* and other first-person shooters have them as well, as do 2D games. In the original *Legend of Zelda*,

for instance, when you have Link walk off one side of the screen, he appears on the other side of a new screen in another part of the large virtual space.

Video games weren't born with these extra-large virtual spaces, though. *Pong*, *Spacewar*, *Space Invaders*, and *Asteroids* are a few of the many games that have a single screen as their playing field. The idea of a game with a virtual space bigger than the screen had to be developed and implemented for the first time at some point.³ This was done by Warren Robinett, as he designed and programmed *Adventure*, the first graphical adventure game, for the Atari VCS in 1978.

Engage with *Half-Life 2* and you could find your avatar, Gordon Freeman, surrounded by attacking enemies who provide supporting fire for each other, dodge, and hide behind cover, powered as they are by what the game industry calls *artificial intelligence*, or AI. The pleasure of many solo games, whether they are real-time strategy games such as *Warcraft III: Reign of Chaos* or first-person shooters, comes from the worthy but surmountable challenge that computer opponents are able to provide.

The computer's ability to play against a person and to play somewhat like a person, rather than just serving as the playing field and referee, wasn't a given in the early days of gaming. Early on, most games were either two-player, like *Pong* and *Spacewar*, or else offered an asymmetric challenge, like that of *Space Invaders*. But there were other developments that helped the industry move toward today's crafty computer-controlled enemies. One early example was Alan Miller's Atari cartridge *Basketball*, which, in its 2K of code and graphics, managed to provide a computer-controlled opponent for a one-on-one game. But even before then, one of the VCS launch titles, *Video Olympics*, offered a one-player "Robot Pong" mode that provided an opponent who, although not anthropomorphic, managed to be challenging without being impossible to defeat.

It's obvious to any gamer today, and certainly also to those who produce games, that there are well-established videogame genres: first-person shooters, real-time strategy games, sports games, driving games, platformers, adventure games, and survival horror games, for instance. Video gaming wasn't always stratified in this way. From the very early days, in which two-player head-to-head challenges predominated, video games began to branch out as games employed many types of hardware and software interfaces, display technologies, game forms, and representations. Gradually, conventions of different sorts began to emerge and various genres became evident.

Some of the development of today's videogame genres arose thanks to computer games and arcade games, but games for the Atari VCS made

important contributions as well. Certain genres the Atari VCS helped develop (such as the vertical scroller, which was fostered by Activision's *River Raid*) do not define important sectors of today's videogame market. Others remain influential, such as the graphical adventure game, the prototype of which was Atari's *Adventure*, and the platformer, pioneered in Activision's *Pitfall!* One game critic even traces the origin of survival horror to the 1982 VCS cartridge *Haunted House*.⁴ Regardless of whether the case for this lineage is persuasive, it is obvious that the Atari VCS was at least a seedbed for videogame genres, if not the forge in which many were formed.

The Atari VCS is certainly a retro fetish object and a focus of nostalgia, but it is also much more than this. The system is essential to the history of video games, and in niches it remains a living part of the modern videogame ecology.

Cartridge Games for the Home

The Atari Video Computer System was the first successful cartridge-based videogame console. (In 1982, when the Atari 5200 was introduced, the system was renamed the Atari 2600, the new name being taken from the system's original product number. Because our focus in this book is on the period 1977–1983, we have decided to call the console “the Atari VCS” throughout the book.) The system appeared at a time when the vast majority of video games were played in bars, lounges, and arcades. The arcade cabinet has become a rare sight in the United States, but in their best year, coin-operated games collected quarters that, adjusting for inflation, sum to more than twice the 2006 sales of U.S. computer and videogame software.⁵

Arcade games derive directly from tavern and lounge games such as pinball. They are indirectly descended from games of chance, including midway games and slot machines. Among his many trades, Atari founder Nolan Bushnell worked the midway as a barker before founding Atari.⁶ His contributions to video games owe much to the principles he learned from his experiences at the carnival.

Midway games rely on partial reinforcement—a type of operant conditioning that explains how people become attached (and possibly addicted) to experiences. Partial reinforcement provides rewards at scheduled intervals. Psychologists Geoffrey R. Loftus and Elizabeth F. Loftus make the argument that video games offer superlative examples of partial reinforcement, presenting incentives at just the right moments to encourage players to continue or try again when they fail.⁷

The classic midway games, which involve things like throwing a ball into a basket or knocking down bottles, appear to be contests of skill. But the barker can subtly alter the games to tip the odds in or out of his favor. For example, by slightly, imperceptibly turning the angle of the basket, the basketball game operator can almost ensure failure, or make success very easy, for a particular throw.

Midway games are illusions more than tests of skill, designed to offer the player just enough positive feedback to give the impression that winning is easy, or at least possible. The midway barker must occasionally allow players to win, persuading onlookers and passersby that the game is a sure thing. Bushnell was a natural barker; he had an uncanny ability to read people and play to their weaknesses. He knew that the big, brutish fellow would be willing to drop a small fortune trying to beat a game that he'd just seen a weakling master.

It was as if Bushnell had all of this in mind already when he first started working with video games. As an electrical engineer educated at the University of Utah, he discovered *Spacewar* at school in 1962. That game ran on the PDP-1 minicomputer and displayed simple graphics on an oscilloscope. Steve Russell, an MIT student, had created *Spacewar* earlier that year. The game quickly spread to the few institutions fortunate enough to have a PDP-1. Given the price tag of more than \$100,000, these were usually universities and laboratories.

Bushnell spent the next decade trying to make a version of *Spacewar* simple enough to run on more common, less expensive hardware. The result was *Computer Space*, which arcade game manufacturer Nutting Associates released in 1971 to very limited commercial success. Complexity of play was part of the problem—the general public wasn't accustomed to arcade games. Parlor and midway games inspire play based partly on familiarity and partly on external rewards. To make a breakthrough, Bushnell needed to merge his experience as an electrical engineer and as a midway barker.

Slot machines certainly implement the midway barker's technique, providing scheduled payouts of varying amounts based on complex odds tables. These tables were originally encoded mechanically and are now represented electronically. But pinball machines and video games give the player partial control over an experience, and in that respect they have more in common with midway games than with slot machines. In the taverns that first hosted Bushnell and Al Alcorn's coin-operated *Pong* (1972), the game became a social hub, serving a function that darts, pinball, and related tavern sports had fulfilled in that space. In *Pong* and its siblings, partial reinforcement operates on two registers. First, the

game encourages continued play and rematches—it promotes “coin drop,” a measure of the rate at which a machine takes in cash.⁸ Second, the game encourages players to remain in the bar, ordering more food and drink. It is important to the history of video games that they bring their persuasive powers to bear within specific architectural spaces, enticing players to enter and remain within certain places.

As tavern culture gave way to the video arcade of the late 1970s and early 1980s, secondary pursuits like eating food surrendered to the primary pursuit of playing games. Arcades had more in common with casinos than with taverns. Bushnell, ever the entrepreneur, recognized this as a market opportunity and decided to create an arcade space with the additional social and gastronomical goals of a tavern, one that would also appeal to a broader audience. While still at Atari, he hatched the idea for Chuck E. Cheese’s Pizza Time Theatres, a place for kids and families to eat pizza and play games.⁹ Here, Bushnell combined all of his prior influences. Chuck E. Cheese’s was an arcade: its games encouraged continued play and cross-cabinet play. It was also a restaurant: food and drink drew players to the locale and kept them there longer. Finally, it was a midway: players collected tickets from games of skill and chance like skeeball in the hopes of exchanging them for prizes.

Yet despite Bushnell’s very relevant background, *Pong* was not simply and directly the result of one man’s midway job. In 1958, Willy Higinbotham created a playable version of tennis that ran on an analog computer, with display output to an oscilloscope, just as *Spacewar* would do half a decade later. Higinbotham worked at the Brookhaven National Laboratory, a federal nuclear physics research facility on Long Island. His game, dubbed *Tennis for Two*, was created as a demo for the lab’s annual visitors’ day. Higinbotham intended it both as a distraction from the rather mundane operation of the facility and, purportedly, as evidence of the future potential for nuclear power.

While Bushnell was working on his tavern-grade adaptation of *Spacewar*, Ralph Baer commenced work on his television gaming device, the “Brown Box.” Like Bushnell, Baer saw the potential for computer games among a broader market, but his great equalizer of choice was the television, not the tavern. The Brown Box was eventually commercialized in 1972 as the Magnavox Odyssey, the first home videogame console. Baer, a fervent supporter of patents and intellectual property protection for software and electronics, worked with Magnavox to battle successor technologies in court throughout the 1970s and 1980s in many lawsuits, some of which named Bushnell and Atari as defendants. Some of the claims against Atari rested on the similarity of *Pong* to the Odyssey’s tennis game,

which Bushnell had seen before *Pong* was built. Magnavox prevailed. Baer's opposition to similar-looking work seems somewhat ironic, though, given the similarity between the Brown Box's television tennis game and Higinbotham's *Tennis for Two*.¹⁰

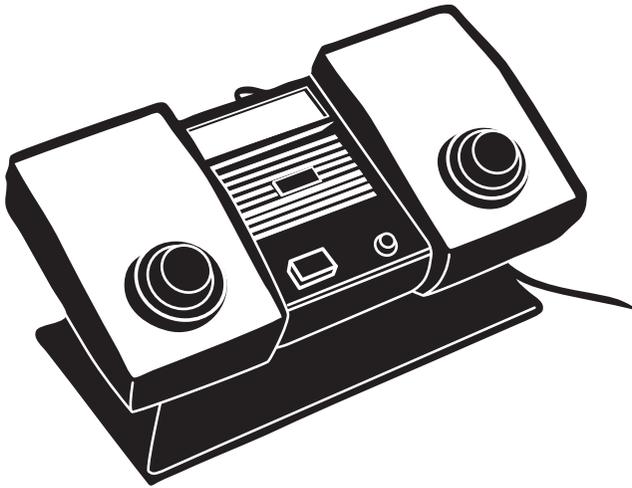
Legal disputes aside, Baer and Bushnell were alike in focusing on one important component in their efforts to create consumer-affordable video games: the television. The Odyssey very obviously relied on the tube in a user's own den or living room, but the arcade game *Pong* was television-based as well, even though most of the TV was hidden away. Al Alcorn, the engineer who built *Pong*, purchased an ordinary consumer-grade black-and-white television for the cabinet, paying much less than he would have for the equivalent industrial monitor.¹¹

The first *Pong* unit was installed in Andy Capp's Tavern, a bar in Sunnyvale, California. Increasingly apocryphal stories of the game's installation report lines out the door but almost never mention the precedent for coin-operated video games in Andy Capp's. When Alcorn installed *Pong* in the summer of 1972, *Computer Space* was sitting there in the bar already.¹²

Pong solved the problem that plagued *Computer Space*—ease of use—partly by being based on the familiar game table tennis and partly thanks to the simplicity of its gameplay instructions. "Avoid missing ball for high score" was a single sentence clear enough to encourage pick-up play, but vague enough to create the partial reinforcement of the slot machine and the midway; after failing, players wanted to try again. One other important sentence appeared on the machine: "Insert coin."

Pong's start in a Silicon Valley tavern rather than a corner convenience store or shopping mall is an important detail of the medium's evolution. Bars are social spaces, and the context for multiplayer games had already been set by the long tradition of darts, pool, and other games common to the tavern. *Pong* was launched in 1972; volume production of the machine started the next year; and, by 1974, there were 100,000 *Pong*-style machines that, as Martin Campbell-Kelly explained, "largely displaced pinball machines, diverting the flow of coins from an old technology to a newer one without much increasing the overall take."¹³ But taverns are also adult spaces that are fewer in kind and number than the millions of living rooms and dens that had access to video games thanks to Baer and Magnavox. At a time when coin-ops ruled the market, part of the appeal of the home console system was its promise to tap into a new market of kids and families.

In 1973, just a year after *Pong*'s coin-op release, Atari started eyeing the home market for video games. The company's home version of *Pong*



1.1 To play Atari's *Home Pong*, the two players each use one of the knobs to control a paddle that appears on the TV screen.

(figure 1.1) boasted considerable technical advances over the Odyssey, including an integrated circuit that contained most of the game's logic on a single chip, on-screen scoring, and digital sound. The device connected to the television directly, but was small enough to store out of the way when not in use. Atari agreed to let Sears sell it exclusively, and the department store initially ordered 150,000 units.¹⁴ Atari's triumph was short-lived, however. In 1976, General Instrument released its \$5 AY-3-8500, a "Pong-on-a-chip" that also contained simple shooting games. This component allowed even companies without much electronics experience to bring *Pong*-like games to market, and many did just that. Campbell-Kelly writes that there were seventy-five *Pong*-like products available by the end of 1976, "being produced in the millions for a few dollars apiece."¹⁵

Even if Atari had cornered the market for home *Pong*, owning the system wouldn't have done anything to directly influence future purchases. Try as Atari did to enhance their product, offering new features and more controllers for multiplayer action in later versions, how many *Pong* units could one house have needed? Those at Atari therefore sought to imitate some features of the nascent personal computer with a home console that used interchangeable cartridges, allowing the system to play many games. There would be an important difference from home

computing, though: all of the cartridges for the system would be made by one company.

The tremendous success of *Pong* and the home *Pong* units suggested that Atari should produce a machine capable of playing many games that were similar to *Pong*. The additional success of *Tank* by Kee Games (a pseudo-competitor that Atari CEO Bushnell created to work around the exclusivity that distributors demanded) suggested another similar game that the cartridge-based system should be capable of playing. *Tank* featured two player objects, each controllable by a separate human player, along with projectiles that bounced off walls. The computational model and basic game form were almost identical to those of *Pong*, and became the essence of *Combat*, the title that was included with the original VCS package. The simple elements present in these early games would be the basis for the console's capabilities from that point on.

Previous attempts at home machines that used interchangeable cartridges, such as the Magnavox Odyssey and the Fairchild Video Entertainment System (VES)/Channel F, brought to light some potential benefits and risks for such a system. Baer's Odyssey, released in 1972, played twelve games, but the players of these games had to attach plastic overlays to the screen to provide the sort of background that would later be accomplished with computer graphics. The machine had no memory or processor. Although the experience of playing the Odyssey was certainly that of a video game, and was important in fostering the market for home video games, the system was perhaps too simplified, even for the time. Playing it may have seemed closer to board game play with a television supplement than to later video gaming. (The inclusion of play money and dice with the system couldn't have helped in this regard.) From the release of the Odyssey in 1972 until it was discontinued in 1975, it seems that between 200,000 and 350,000 units were sold.¹⁶ The machine introduced home videogame systems to the world, but not on the scale that the Atari VCS would, beginning in the late 1970s.

Fairchild's VES, released in 1976, was the first programmable, interchangeable cartridge system. It sported an onboard processor and random-access memory (RAM). The system had a rapid name change when Atari's VCS was released, and is better known today as the Fairchild Channel F. Even before Fairchild's system was market-tested, though, Warner Communications purchased Atari. The purchase was motivated primarily by the commercial promise of an extensible home console.¹⁷ This 1976 acquisition provided the capital that Atari needed to bring the Atari VCS to market.

Design of the Atari VCS

The engineers developing the Atari VCS needed to account for two goals—the ability to imitate existing successful games and some amount of versatility—as they designed the circuitry for a special-purpose microcomputer for video games. Material factors certainly influenced the design. At one extreme was the high cost of hardware components. The Channel F was manufactured by Fairchild Semiconductor, and unsurprisingly the system used that company's Fairchild F8 CPU, a specialty processor created by future Intel founder Robert Noyce. At the other extreme was a lack of flexibility. The Odyssey's games were implemented directly in diode-transfer logic (DTL) on the console's circuit board. The cartridges for the Odyssey, unlike those for the Fairchild system, simply selected a game from a set of hard options.¹⁸ The Atari VCS would need to navigate between the Scylla of powerful but expensive processors and the Charybdis of a cut-rate but inflexible set of hardwired games.

It could be done. In 1975, MOS Technology had released a new processor—the 6502. At the time, the chip was the cheapest CPU on the market by far, and it was also faster than competing chips like the Motorola 6800 and the Intel 8080.¹⁹ The 6502's low cost and high performance made it an immensely popular processor for more than a decade. The chip drove the Apple I and Apple II, the Commodore PET and Commodore 64, the Atari 400 and 800 home computers, and the Nintendo Entertainment System (NES). It is still used today in some embedded systems.

This chip seemed attractive, as cost was the primary consideration in the design of the Atari VCS. The system needed to be much more affordable than a personal computer, which was still a very rare and expensive commodity. When Apple Computer released the popular Apple II in 1977, it cost \$1,298, even after Steve Wozniak's many cost- and component-saving engineering tricks. The same year, Atari released the VCS for \$199.²⁰ The price was just above the console's manufacturing cost, a common strategy today but an unusual one in the 1970s. Atari was betting heavily on profiting from cartridge sales, as it indeed would do.

In 1975 Atari acquired Cyan Engineering, a consulting firm.²¹ Cyan's chiefs, Steve Mayer and Ron Milner, were the ones who selected the MOS 6507 for the VCS project. This chip was a stripped-down version of the already inexpensive 6502. From a programmer's perspective, the 6507 behaves more or less identically to a 6502, but it cannot address as much memory, a limitation that ended up affecting the maximum capacity of videogame cartridges for the system.

The 6507 was available for less than \$25; similarly capable Intel and Motorola chips went for \$200.²² But the 6507 CPU was only one component—the Atari VCS still needed additional silicon for memory, input, graphics, and sound. The CPU does the essential arithmetic at the core of computation, but a videogame system also needs to carry out other functions; among them, the important job of producing sound and graphics. At the time, computer graphics were mostly managed in read-only memory (ROM). Character sets and video memory for a grid of rows and lines of text were reserved in a special space in ROM chips on the motherboard. Such was the case for the Tandy TRS-80 and Commodore PET, both also released in 1977. The Apple][’s graphics and sound system was implemented in a similar but more sophisticated way, thanks in part to Steve Wozniak’s experience designing an Atari arcade game. As Wozniak explained:

A lot of features of the Apple][went in because I had designed *Break-out* for Atari. I had designed it in hardware. I wanted to write it in software now. So that was the reason that color was added in first—so that games could be programmed. I sat down one night and tried to put it into BASIC. Fortunately I had written the BASIC myself, so I just burned some new ROMs with line drawing commands, color changing commands, and various BASIC commands that would plot in color. I got this ball bouncing around, and I said, “Well it needs sound,” and I had to add a speaker to the Apple][. It wasn’t planned, it was just accidental.²³

Wozniak engineered capabilities into ROM, burning what he needed onto chips that went onto the motherboard. Each additional chip meant more cost—exactly the luxury that the ultra-low-cost Atari VCS couldn’t afford.

The Atari VCS needed a system for graphics and sound similar in principle to Wozniak’s flexible Apple][, but simpler in its design and having less of an impact on hardware costs. For this purpose, the Atari VCS used a custom chip, the Television Interface Adaptor (TIA). Joe Decuir and Jay Miner designed the TIA, which was code-named “Stella”—a name also used for the machine as a whole, and one which came from the brand of Decuir’s bicycle.²⁴ Of course, the two sought to simplify the hardware design as much as possible, reducing its complexity and cost. For this reason, a custom graphics chip was the only real option. The cost of TIA research and development must have far outweighed any other development cost for the system, and yet it was a wise investment, given that graphics and sound are so essential to video games.

RAM, the memory programs use to store temporary information while they are running, was very expensive at the time, so an important cost-saving measure was limiting its use. In 1977, the Apple II shipped with 4K of RAM. The TRS-80 and PET, both shipped in that same year, also sported 4K. In 1982, the Commodore 64 shipped with 64K, the maximum amount addressable by the 6502. By contrast, the Atari VCS has only 128 *bytes* of RAM. That amount is $\frac{1}{32}$ of that in general-purpose microcomputers of the late 1970s and not enough to store this ASCII-encoded sentence. RAM remained a costly prospect through the 1980s, and many home game consoles scrimped on it to reduce costs. The 128-byte memory of the VCS was twice as large as the standard RAM of the Channel F. The significantly more advanced NES had sixteen times as much RAM at 2K.

In addition to the processor, the graphics system, and memory, the fourth major component of the system, a chip called the Peripheral Interface Adaptor (PIA) or RAM/Input/Output/Timer (RIOT), handles input from the two player controls and the console switches. Unlike the Odyssey, the Channel F, and a competitor that was to arrive in 1979 (the Intellivision), the VCS let players plug in different controllers right out of the box. Two different kinds were included with the system, and several different styles were marketed by Atari and third parties during the console's long commercial life. Most notably, though, the Atari VCS introduced the joystick to the home videogame player as the standard control.

In studying the Atari VCS from the perspective of the platform, several things stand out about the system and its influence on the future of video games. One is the strong relationship between the console and the television. Baer correctly predicted that the TV would be central to video games. (Games driven by computer power had previously been designed for less common displays, such as oscilloscopes, or crafted for use on print or video terminals.) The Atari VCS—particularly its graphics and sound chip, the TIA—is designed to interface solely (and weirdly) with a standard CRT television, the sort common in living rooms and dens of the 1970s. Its controllers and peripherals were fashioned for use on the floor or the couch. The games made for the platform are likewise oriented toward home use—either for enjoying the arcade experience at home or for playing in different ways with friends and family. The focus on the production of images for display on the TV helps explain why games running on circuits and later computers became known as “video games.”

Another strong current in the work on the Atari VCS is the powerful influence of earlier games. Many—perhaps most—VCS cartridges are to a greater or lesser extent ports of arcade games. The system's architecture

was designed with the popular coin-op games *Pong* and *Tank* in mind. Many early VCS titles were directly ported from coin-op games. Even very innovative titles like *Adventure* were directly inspired by games on other computer systems. After 1980, licensed arcade and film adaptations became popular as well, especially at Warner-owned Atari. But beyond ports of coin-op games and adaptations from other entertainment media, the 1977–1983 era was one of uncertainty and experimentation in video games. Nobody really knew what would make a good subject for a game, and many relied on previous successes. Atari’s liberal use of the term “video” in game titles underscores the company’s reliance on transforming familiar subjects into games for play on a television: *Video Olympics*, *Video Checkers*, *Video Cube*.

A final observation is the tremendous representational flexibility of the machine and the less-than-obvious reason for this flexibility. The games created for the platform during its long life cover innumerable subjects and situations: dogfighting, bridge, hockey, treasure hunting, lassoing, slot car racing, dental care, and even sex acts. The breadth of the system’s software library becomes even more striking when one considers that two simple arcade games were the major inspirations for its hardware design—and that no one fathomed how successful and long-lived the console would be.

So much was possible on the Atari VCS, and not because it was a powerful computer. It wasn’t powerful at all. Rather, so much was possible because the machine was so simple. The very few things it could do well—drawing a few movable objects on the screen one line at a time while uttering sounds using square waves and noise—could be put together in a wide variety of ways to achieve surprising results.

Plan of the Book

In this book we concentrate on six VCS cartridges while also discussing many others along the way.²⁵ We selected these six cartridges from the many hundreds that have been developed because they particularly enlighten the discussion of the VCS platform and creative production upon it. We discuss them in chronological order, so that the development of programming practices and the changes in home and arcade video gaming can be tracked more easily through time. The cartridges that are central to our discussion are as follows:

- *Combat*, the cartridge that was originally bundled with the Atari VCS. This set of two-player tank and plane games demonstrates almost all

of the basic hardware capabilities of the system in a straightforward way. It also reveals a great deal about the relationship of home video gaming to arcade games, showing how even the first home games based on arcade games would use them as a starting point and often transform them.

- *Adventure*, which established the action-adventure genre. This game represents a virtual space that is larger than the screen, showing how some of the affordances of the VCS platform can be used for purposes that were different than those originally intended. *Adventure* was also a radically different adaptation of an all-text computer game, one that again helps to reveal the influence of platforms in creative production.

- *Pac-Man*, a more direct take on a successful arcade game—one that spawned a massive craze, or, one might say, a fever. The cartridge was widely derided as being a poor, inauthentic port of the arcade game, yet it nevertheless became the overall best-selling cartridge for the Atari VCS. The mismatch between the arcade *Pac-Man* and the capabilities of the VCS hardware is particularly revealing.

- *Yars' Revenge*, Atari's best-selling original VCS game. Even this "original" started as a conversion of an arcade title. The conversion was jettisoned and a highly complex game took shape, one with unique graphics that was well-suited to home play. This cartridge shows another important way in which some arcade platforms (those using a fundamentally different display technology, *vector* or *XY* graphics) differed from the Atari VCS. It also reveals how a programmer used his knowledge of the VCS hardware to fashion a novel and effective game rather than implementing a partial and ineffective re-creation. The cartridge *Asteroids*, a contemporary of *Yars' Revenge*, is discussed in some detail as another case of a vector graphics game being adapted to the Atari VCS.

- *Pitfall!*, another innovative original that was developed at Activision, the first third-party videogame company. This game helps to show the difference between cartridges produced by Activision and those produced by Atari for the same platform, and it also provides a way to look at the rise of third-party videogame companies and the platform-related challenges they faced. *Pitfall!* was also a critical step in the development of the action-adventure genre and an important step toward the side scroller.

- *Star Wars: The Empire Strikes Back*, probably the most unusual choice of these six. Obviously, it is a game that was produced under license and was meant to exploit the success of a popular film. Although not the most famous VCS game of this sort, *Star Wars: The Empire Strikes Back* shows how a compelling cinematic situation can be translated effectively into a videogame challenge. The cartridge also provides a good opportunity to

discuss the explosion of third-party titles and the interaction between media properties and video games, along with the collapse of the Atari VCS market that ensued in 1983. Finally, it reveals how much the use of the VCS platform had escaped from the proprietary hold of Atari and how much it had advanced during the time period we are considering.

The book concludes with a short chapter. Although our platform study focuses on the Atari VCS during 1977–1983, this final chapter briefly considers some of the high points of the life of the Atari VCS from 1983 to the present, discussing what else has been learned about the platform and how this platform has interacted with culture during this span of time. Finally, in an afterword, we look ahead to see what other platforms, and what other issues, can be addressed by future platform studies, and to consider what insights this approach can offer as we continue to think about creative digital media.

