Game On: Atari Video Computer System – Bring the story home

Joe Decuir, IEEE Fellow
IEEE CES Board of Governors 2015-2017
CES Distinguished Lecturer 2013-2016
Agenda

- Bring the story home
- Early video games
- The market problem to solve in 1976
- How the 6502 changed things
- Turning a prototype into a product
- Game design: Tank
- Brilliant game designers
- What we learned
- Resources
Entertainment: in the network, and then at home
Entertainment in 1970

When I started the college, the stories were all controlled by the public media companies:

- Movie companies made films and showed them in theaters
- TV networks provided news and assorted shows, punctuated by commercials, which we could watch at home
- Musicians recorded songs, which we could listen to on the radio or on vinyl records at home
  - I was lucky enough to see some of them in concert
- A few of us also made it to live theaters (with actors)
- We – the customers - did not control the story
First Video Games

Ralph Baer was a pioneer, recognizing that it was possible to bring entertainment home.

He imagined a machine which allowed electronic gaming on a “Brown Box” in a family home.

Ralph was unlucky – he worked in defense.

His employer licensed his design to Magnavox – as the Odyssey.

Ralph Baer in his basement lab, 2014
Video Games in arcades

- Video games were known from experiments on university computers.
- Entrepreneurs like Nolan Bushnell experimented with bringing the gaming experience to arcades, starting with Computer Space.
- Nolan was lucky to witness a demonstration of the original Magnavox Odyssey (Brown Box).
- Al Alcorn re-implemented the game idea as an arcade game, named Pong.
  - This was wildly successful.
- In 1975, they brought out a home version.
  - another success.
Microprocessor Revolution

- In the mid-20th century, computers were large and expensive
  - Recommend: visit Computer History Museum

- Major steps:
  - Big computers became commercial – IBM et al
  - Digital Equipment et al created ‘mini-computers’

- 1971: the first microprocessor appeared
  - Intel 4004 handled 4 bits at a time, 50kHz bus
  - A 4004 + a 4001 (ROM & IO) = minimum system

- 1974: we had the Intel 8008, w/8 bit memory

- 1975: added the Intel 8080 and the Motorola 6800
The Problem to Solve
Atari’s Market Problem

- They had been leading the arcade game business, starting with Pong
- They had extended their position with complex arcade games, like Tank
- They had succeeded competing with Magnavox for home paddle games
- The clear next move: bring complex games home
- Atari knew they had competition everywhere
Implementation Choices

- Design new custom application specific integrated circuit (ASIC) for each game
  - Pong for home use is an example
- Design a programmable system, using a new microprocessor, and an ASIC to drive the screen and the audio
- In 1974-1975, microprocessors and memory were expensive!
A bit mapped design

- Graphic systems at the time would have a bit map or character map.
- A common graphics terminal would have a 64x16 character array, backed by a ROM for character graphics.
  - A 1K byte character RAM cost close to $10.
- A pure bit map would become prohibitively expensive. On a monochrome monitor, 512x256 would be 16K in DRAM.
The default screen map design

- Given the processors and memory in mid-1975, a programmable architecture looked too expensive.
- A display of 160 x 96 x 2 bits would need a 4K x 8 DRAM array, perhaps $25 dollars.
  - Not much RAM left for game variables.
- A processor like the 6800 was quoted around $25 in large quantity.
- An ASIC would be needed to manage the RAM and generate the display.
Then MOS Technology changed the analysis.
6502 introduction

- A team led by Chuck Peddle left Motorola and teamed up with MOS Technology to build a better microprocessor

- Big changes:
  - Replace enhancement mode NMOS by depletion load NMOS
  - Adjust the register architecture to define better memory addressing
  - Adjust the architecture for little-endian address arithmetic pipelining
How this changed the analysis

- The 6502 was priced at $25 per sample
  - I bought one in September 1975 at Wescon
  - So did Wosniak and Ron Milner of Atari
- MOS Technology quoted $5/each in large quantities
- Depletion load NMOS made the parts fast enough for 1-2 MHz memory.
  - Median 6800s could handle 500 kHz memory
- The architecture changes were decisive
Address arithmetic

- Data processing involves manipulating addresses
- The 8080 had a 16 bit H and L register pair
  - Computing an address was intensive
- The 6800 has some address computation
  - It was limited, a 16 bit X register was offset with an immediate 8 bit value
- The 6800 had a nice trick: the first 256 ‘zero page’ memory locations could be addressed with shorter instructions
6502 memory addressing

There were two innovative standard addressing modes for most ALU instructions:

- Indexed indirect addressing
  - 8 bit X register added to 8 bit zero page offset. This selects a 16 bit memory address.

- Indirect indexed addressing
  - 8 bit zero page offset selects a 16 bit memory address. The 8 bit Y register is added to that 16 bit memory address (next slide).
**Indirect indexed addressing**

<table>
<thead>
<tr>
<th>Bus clock</th>
<th>Address bus</th>
<th>Data Bus</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PC</td>
<td>Op code</td>
<td>Load instruction using indexed indirect</td>
</tr>
<tr>
<td>2</td>
<td>PC+1</td>
<td>ZPA</td>
<td>8 bit value pointing to a base address in zero page</td>
</tr>
<tr>
<td>3</td>
<td>ZPA</td>
<td>DAL</td>
<td>Fetch the Data Address Low byte</td>
</tr>
<tr>
<td>4</td>
<td>ZPA + 1</td>
<td>DAH</td>
<td>Fetch the Data Address High byte</td>
</tr>
<tr>
<td>5</td>
<td>DAH:DAL + Y</td>
<td>-data-</td>
<td>Add the Y index register to the DAL+DAH to generate the effective address</td>
</tr>
</tbody>
</table>
Consequence of 6502 speed

- The 6502 was fast enough to compute and write graphics on a line-by-line basis.
- IT DID NOT NEED A FRAME BUFFER!
  - This saved a lot of money.
- The 6502 could run at 1.2 MHz, for 76 memory cycles per TV horizontal line.
  - This would be maybe 25 instructions/line.
- Ron Milner and Steve Mayer of Atari build a prototype (next slide)
  - Great luck: I was hired to debug it
Atari VCS proof of concept

Computer History Museum
IEEE Consumer Electronics Society

Generating a Product
Assembling the team

- The prototype and I were sent to Los Gatos
- I was apprenticed to Jay Miner (next slide)
- Jay worked for Bob Brown, who in turn worked for Al Alcorn – engineering VP
- We assembled a team to build the system: Wade Tuma, Neils Strohl, etc.
- We had ASIC chip designers and testers
- We needed game designers. We hired Larry Wagner. Larry in turn hired Dave Crane, Al Miller, Larry Kaplan and Bob Whitehead.
Jay Miner
General plan

- We looked at the prototype as an existence proof. As Brooks wrote (Mythical Man-Month) we ‘plan to throw the first one away’

- We did a new system design, with a CPU, an ASIC, a combo device with memory and IO and an external ROM on a cartridge.
  - We had decided that having RAM to load a program would cost too much

- We designed a depletion load NMOS design

- Then we emulated that design with 74LS TTL
System fundamentals

- We kept the essentials:
  1. A 6507, a cost reduced 6502 in a 28 pin package, losing some functionality, but saving 50 cents.
  2. A 6532, with 128 bytes of RAM, 16 IO ports and a programmable timer
  3. The “Television Interface Adaptor”, with video, audio and six extra specialized input ports
  4. A 2K or 4K byte ROM on an inserted cartridge.
Atari VCS system diagram
Sound circuits

- Two sound circuits, one per player
- Three components for each:
  1. Frequency divider
     - Start with 2x horizontal rate ~ 31 kHz
     - Add a 5 bit prescaler
  2. Noise generator: 4 bit, 5 bit and 9 bit polynomial counters
     - 4 bit poly counter made tank noises
     - 9 bit poly counter made jet noises
  3. Multiplying 4 bit A/D converter
Parallel IO

- 5 6532 IO ports for the game console controls: game start, game select, difficulty (left and right) and color/BW. The 6th switch was power.
- 4 6532 IO ports each for the two controllers
- 2 TIA paddle port inputs for each of the two controllers; we could play 4 player pong games.
- 1 TIA trigger port for each of the two controllers; they could also support a light gun
HW Optimization: collision registers

- In games, objects ‘collide’ with other objects.
- Detecting collisions in software is difficult. The program needs to construct the outline of each object and compare locations.
- It was much easier to add a set of 15 hardware collision registers – simple RS FF.
- The programmer reads and clears them.
- The hardware sets them if any pair of objects and playfield overlap.
SW Optimization: CHRST device driver

- Atari designs used low resolution ‘playfield’ and high resolution moving ‘players’ (sprites).
- Players and ‘missiles’ could have their horizontal positions reset, and be moved a few pixels; they did not have binary position registers and comparators.
- We created a standard subroutine called “compute horizontal reset”. The game designer called it with a sprite index and an 8 bit offset. The subroutine positioned the sprite.
Game controllers

Bundled with the machine:
- Two-axis digital joystick with a fire button
- Dual potentiometer controllers, each with 1 button

Shipped with special games:
- Grey coded 360 degree rotary controllers, bundled with a driving game.
- Keyboard controllers: 3 x 4 key array, bundled with a Basic Math cartridge.
Program flow

In vertical blank time:
- Processing console controls
- Detecting and processing collisions
- detecting any other changes (e.g. timeouts)
- update game status.

In display time:
- Driving the TV display on a line-by-line basis, in what we called the ‘kernel’
- In a following illustration, there are two: score and game field
Game state

- Shape of the playfield
- Location, orientation and shape of two complex (8 bit wide) moving objects: ‘players’
- Location and orientation of 3 simple (1 bit) moving objects: ‘missiles’ and ‘ball’.
- Current settings for 2 sound channels
- Other displays, e.g. score for each player
Translate the state for the kernel

- The program would then compute new values to display, and locations.
- It would set up the zero page pointers that the kernel would use to drive the display after the vertical blank time ended.
- The program would poll the hardware timer to know when to halt game state changes and enter kernel execution.
- The efficiency of the 6502 to compute addresses into the ROM on the fly was crucial for the kernel.
Program flow

Horizontal Line – 228 color clocks

Horizontal Blank – 68 cc
Horizontal Display – 160 color clocks

Vertical Blank & Sync: 54 lines
Vertical Blank processing:
- Process user inputs
- Detect and process collisions
- Compute new game state, including score

Vertical Display: 208 lines
kernel processing:
- compute game graphics pointers,
- load TIA registers
- [option: poll potentiometer inputs]

Vertical 262 Lines Frame

- Process user inputs
- Detect and process collisions
- Compute new game state, including score
Prepare for introduction

- The TIA ASIC worked the first time.
- We built many more prototyping systems to keep the game designers tooled up.
- We had to do a system design, with RFI shielding (aluminum casting)
- We had to tool up with a new factory
- We needed production test equipment and tooling
- One of the reasons we sold out to Warner: we needed the money to afford all this
Launch game titles - 1977

- Air Sea Battle (Kaplan)
- Basic Math (Palmer) – keyboard controllers
- Blackjack (Whitehead)
- Combat (Decuir & Wagner) – bundled game
- Indy 500 (Riddle) – driving controllers
- Starship (Whitehead)
- Street Racer (Kaplan)
- Surround (Miller)
- Video Olympics (Decuir)
Competition: Fairchild Channel F

- This machine was developed at the same time as the Atari VCS
- It illustrated problems with bit mapped design.
- All graphics were in a 128 x 64 pixel bitmap
- All moving objects were drawn in the common bit map
  - Object graphics were very crude
  - Motion was also crude, and slow
- Total production is estimated as 250K units
Production history and lessons learned
We were nicely surprised

- The game designers turned out to be much more creative than the system designers anticipated.
- We made the hardware cheap, to be economically feasible in 1977.
- To do so we moved functionality from the hardware to the firmware.
- The game designers then took advantage of the ability to ‘follow the beam’ and redefine the display on the fly.
Explosion in games

- Over a thousand game cartridges were defined for this game console system.
- The most important ports from arcade games were Pac-Man, Missile Command, Space Invaders and Asteroids.
- The list of original games is very long
- My highlight set includes Adventure (Robinett), Pitfall (Crane), River Raid (Shaw) and Yars’ Revenge (Warshaw).
Production history

- Production started in July 1977, with 250,000 units
- Production stopped in 1989 (Atari VCS jr), with over 30 million units shipped
  - the Nintendo NES replaced the Atari as the leading video game console.
- You can still buy them today!
  - The Flashback 2.0 is based on a new merged design, and 40 built in games.
  - The Flashback 5.0 is based on an emulator, with 92 built in games
1977 ‘heavy sixer’ & 2014 Flashback 5.0
New program development

- At Classic Games conferences, companies like AtariAge support new cartridge development.
  - Several new games are created every year.

- In 2010, Ed Fries, contributor to the Microsoft X-Box game console, created Halo 2600 for the Atari.
  - 33 years after the machine was first shipped

- The Halo 2600 graphics are a lot simpler, given the hardware, but the essence of the game play survives.
Lessons learned

- We put the definition of the display in the hands of the game designers, who were smarter than we hardware designers expected.
- Our conclusion was that our second system should not simply be a bit map and a processor.
- We had created a platform for the art of others, and we planned to build on that for the next system.
- We thought we needed to move fast. The Atari PCS came out two years later, in 1979.
To learn more

- MIT Press published a Platform Series book on the Atari VCS, titled Racing the Beam. It covers Stella, Combat, Adventure, Pac-Man, Yars’ Revenge, Pitfall, Star Wars (game) and the first video game crash.
- The book Atari: Business is Fun is the definitive social history of Atari at the time the VCS was developed. It covers the arcade side of the business as well.
- Warren Robinett is working on a book about game design, titled “The Annotated Adventure”.
- For more technical information the Atari VCS, see the TIA Schematics and the Stella Programming Manual.
- I wrote an article with much more detail in the July 2015 issue of Consumer Electronics Magazine.
My contact information

Joe Decuir
Issaquah, WA USA
IEEE: jdecuir@ieee.org
IEEE Fellow
IEEE CES Board of Governors: 2015-2017
IEEE Global Humanitarian Technology Conference: chair 2015, ieeeghtc.org
Work: joe.decuir@csr.com
Standards Architect, CSR Technology
Chair: Bluetooth Internet Working Group
Vice chair: Smart Environment Study Group
Questions?