# Amiga 1000 Hardware<br>Early History and<br>System Architecture Overview

Joe Decuir & Ronald Nicholson

# Agenda

Introduction

Amiga Business Background

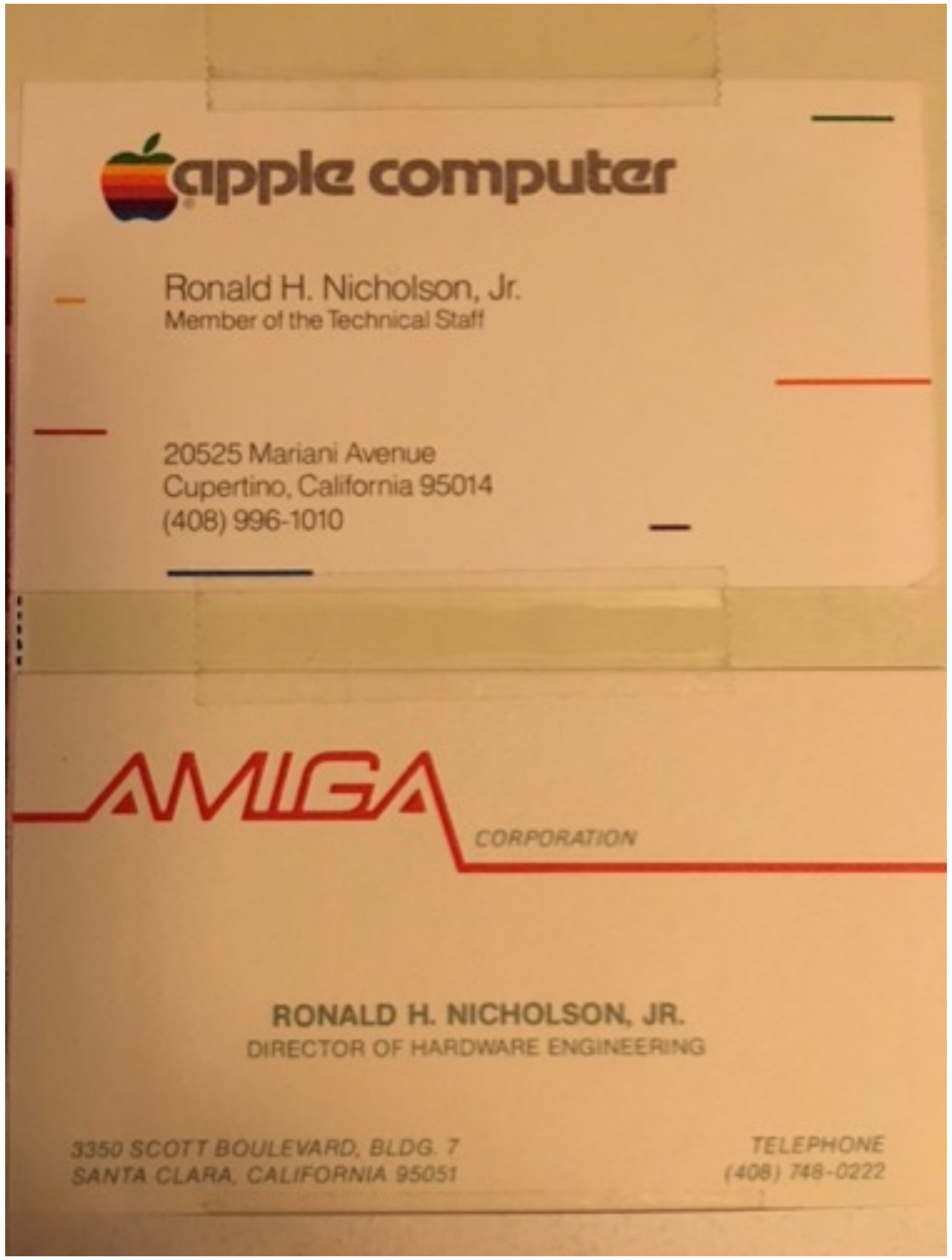
Technological Environment during the Amiga Design

Marketing Goals

Some Key Features of the Amiga Hardware

Summary

# Old Business Cards



apple computer

Ronald H. Nicholson, Jr.
Member of the Technical Staff

20525 Mariani Avenue
Cupertino, California 95014
(408) 996-1010

AMIGA
CORPORATION

RONALD H. NICHOLSON, JR.
DIRECTOR OF HARDWARE ENGINEERING

3350 SCOTT BOULEVARD, BLDG. 7
SANTA CLARA, CALIFORNIA 95051

TELEPHONE
(408) 748-0222

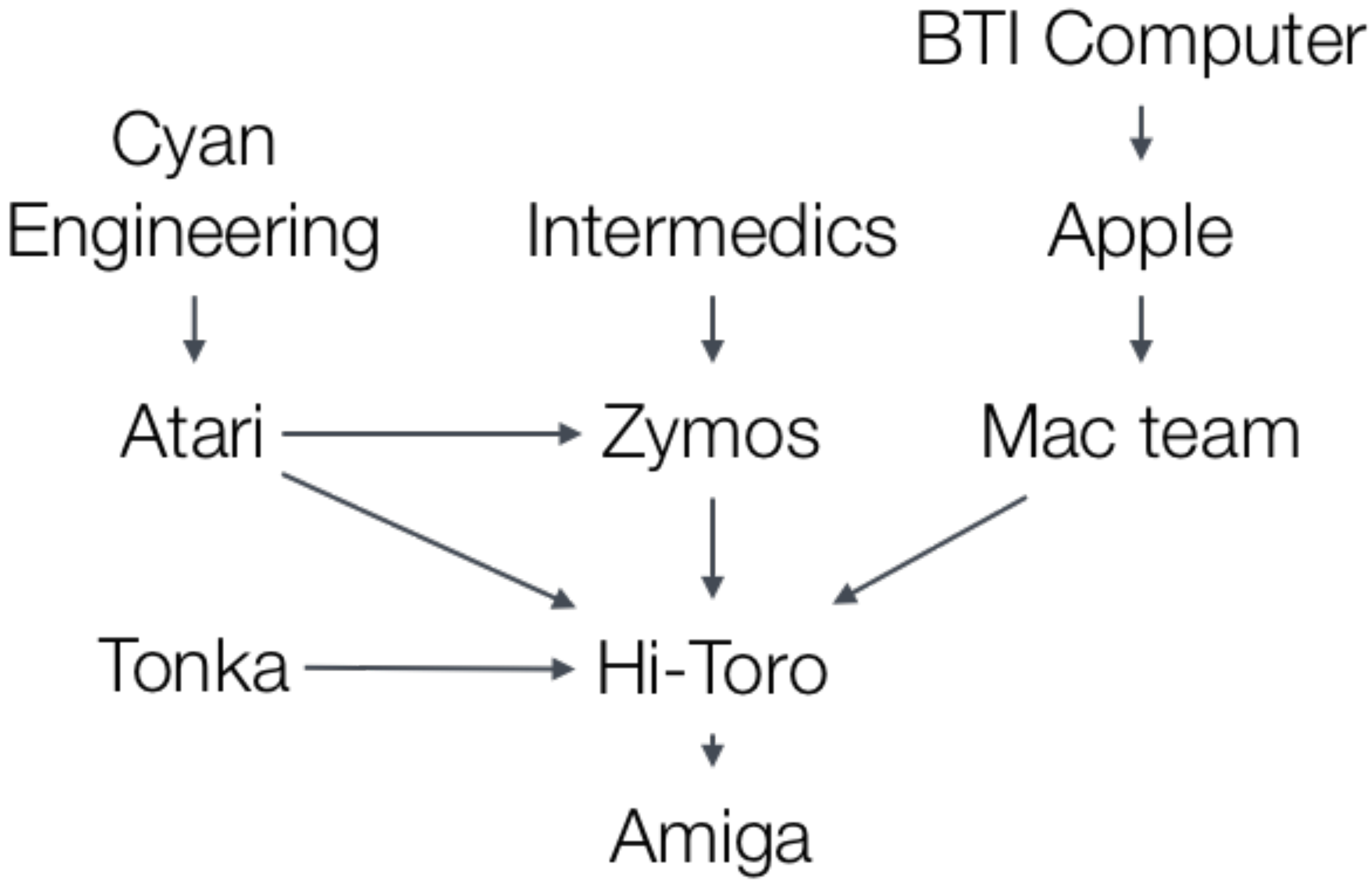
# Who?

Jay Miner, Joe Decuir & Ronald Nicholson are named co-inventors on 4 patents that apply to the Amiga:

- 4,777,621: Video game and personal computer, including genlock
- 4,874,164: Personal computer apparatus for block transfer of bit-mapped image data
- 5,103,499: Beam synchronized coprocessor
- 5,594,473: Personal computer apparatus for holding and modifying video output signals

Joe Decuir worked with Jay Miner on the Atari 2600 and Atari 400/800 Personal Computer

# Mini Genealogy



# Intermedics, Zymos and Funding for a Game Machine

- Intermedics developed implantable pacemakers
- Zymos was funded to make custom pacemaker chips for Intermedics, but that wasn’t a high enough volume product to pay for a semiconductor fab line
- Burt Braddock looks for a high volume chip opportunity, and puts some Intermedics/Zymos funders in contact with Dave Morse

- Zymos’ CMOS semiconductor process turns out to be too slow for a high performance game machine

# Joe Decuir

Atari recruited Joe Decuir to apprentice to Jay on the Atari Video Computer System (aka Atari 2600).

He used to go to work thinking ‘They are paying me to do this?’

First job: debug the concept prototype at Cyan Engineering

The first result: Atari Video Computer System

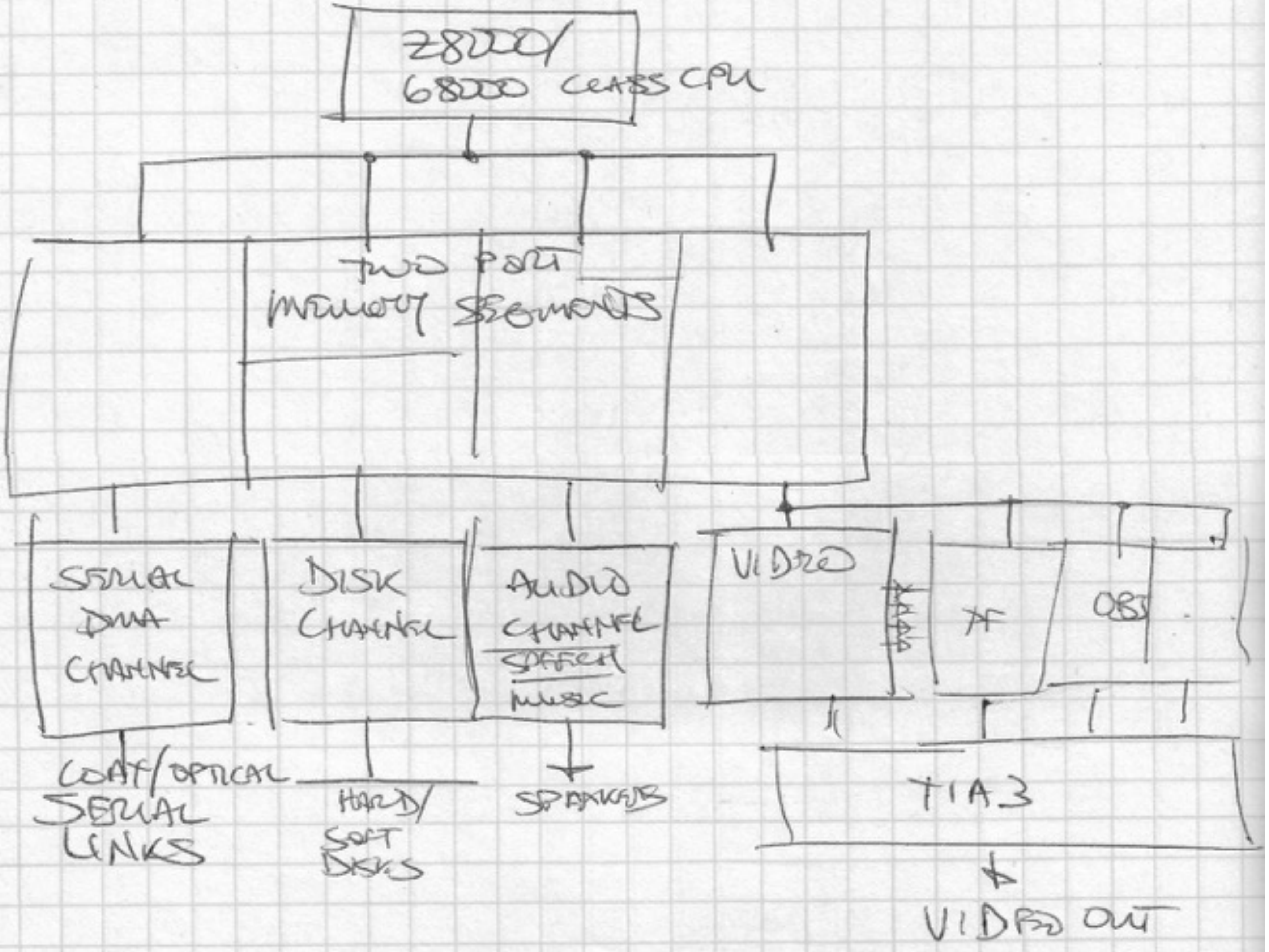
- Joe was understudy, designing parts of the TIA chip, and the Stella system
- Side job: writing the Combat display engine, writing Video Olympics

Second job: Atari Personal Computer System

- Joe Decuir was the system architect
- He devised the three ASIC system: ANTIC, GTIA and POKEY
- The ASICs were a preview of the Amiga (next two slides)

PROPOSAL FOR 'STEPS OUT'

HIGH POWER ENTERTAINMENT COMPUTER



# Prior to Amiga

Compression Labs

BTI Computer Systems

Apple II Peripheral Group

- Super-Serial Card
- Apple II video timing The
- Apple II Unified The The Interleaved Access
- PWM & S.A.M. card
- Bit-blit animation in 6502 code (Bill Budge)
- Lisa mouse pointer

Macintosh custom IWM & CMOS clock chip project

Steve Jobs introduces me to Jay Miner

Apple contracts with Zymos to do a semi-custom ASIC clock chip (2nd source?)

Jay Miner introduces me to Dave Morse

# Existing Technology

Atari VCS (sprites, beam following)

Apple II+ (TDM Unified memory, software PWM audio)

IBM PC/XT, Apple Lisa (Jan ’83)

Affordable NTSC or PAL Color monitors

Sony CD player (Oct ’82, $1600 AFI)

DRAM costs, 64 kBytes was over $100

4 micron NMOS, hand layout, tape-on-rubylith

48-pin plastic DIP packages

# Marketing Requirements

# Marketing Goals

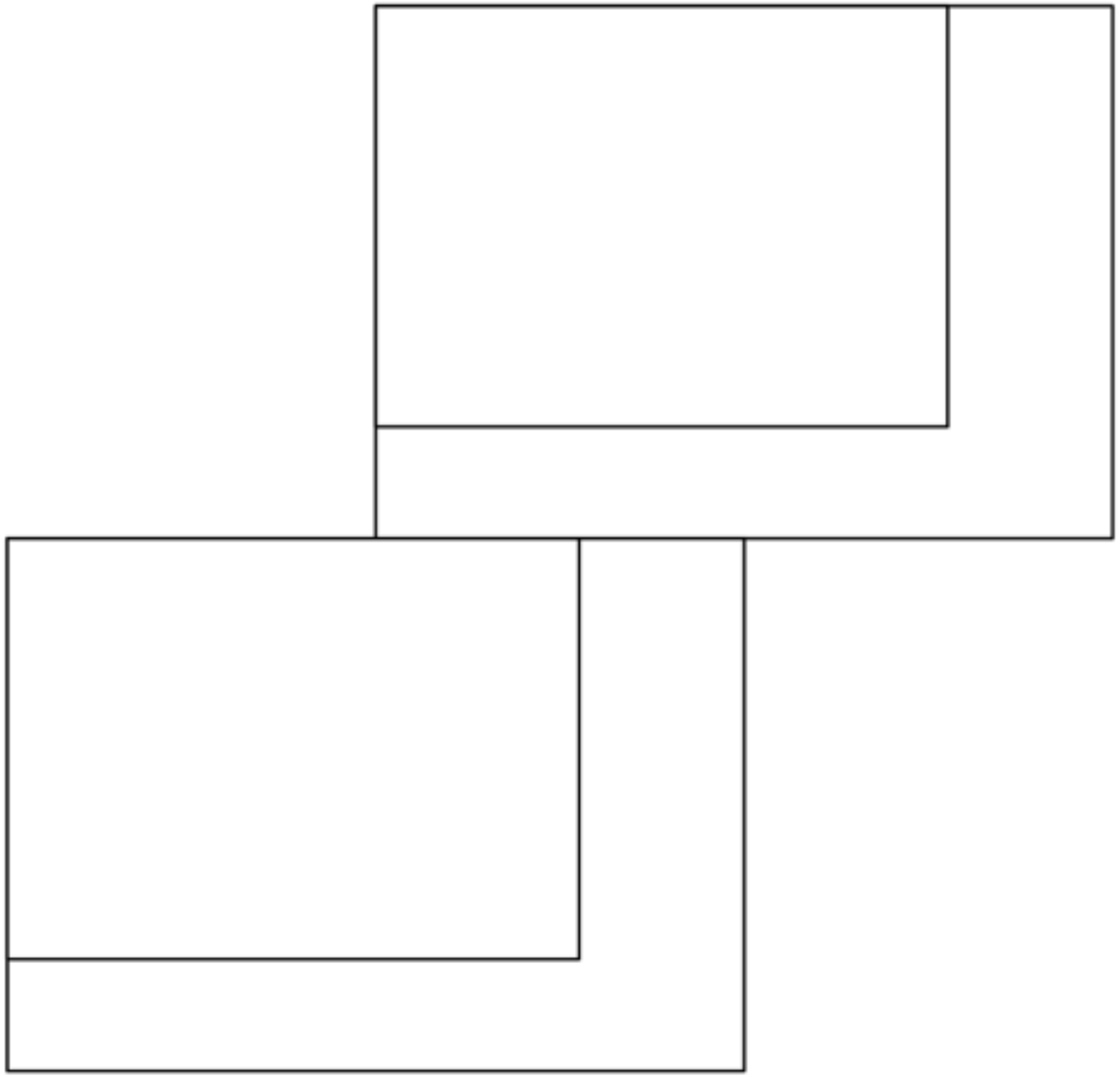
- Dave Morse was the original founder & Hi-Toro CEO
  - Former VP of marketing at Tonka Toys
- Dave's marketing goal was to bring entertainment stories from movies and network TV interactively into the customer's living room
  - Home video game consoles were already doing this
- Dave wanted a machine that could render cartoons in real time
  - We joked about Smurfs
- The target: Invent a way to animate cartoons (given early 1980s technology) at least 10 times faster than existing game machines and PCs could using software rendering
- Jay Miner specified enough expansion options so that the game console could become a personal computer
- We merged state-of-the-art game software animation with a new hardware architecture

# Amiga Hardware Design

# Some Key Hardware Innovations

- Accurate NTSC timing (led to genlock & Toaster)
- DMA driven IO – offloading the processor
- Programmable video co-processor, aka Copper
- Bit-blitter: sprite splicing, area fill, programmable ALU
- Bit-plane video, with flexible priority
- Hold-and-modify as a compressed color encoding
- 4-channel sampled audio synthesis engine
- Configurable interrupt system based on video timing

# 525 line NTSC



# Key Features for Graphics Animation

- Best CPU that was consumer priced: 8 MHz MC68000
- Full NTSC or PAL TV resolution bitmap rendering
- Planar pixel format to simplify Blitter and to fit memory
- Color differential encoding compression (Hold-and-Modify)
- DMA coprocessor that can follow the CRT beam (COPPER)
- ‘Bit-blitting’ inspired by the Xerox Alto and Apple Lisa
- 8 reusable sprite engines

# COPPER – Video Coprocessor

- The Atari VCS game programmers impressed Jay and Joe with their ability to ‘Follow the Beam’ and generate amazing displays
- The Atari PCS ANTIC ASIC had a DMA Display List Processor
- It executed a Display List, which specified the graphics mode on a line-by-line basis: bit maps and character maps
- Each display list instruction could also generate an Interrupt
  - E.g. an interrupt handler could change colors, move sprites horizontally, etc.

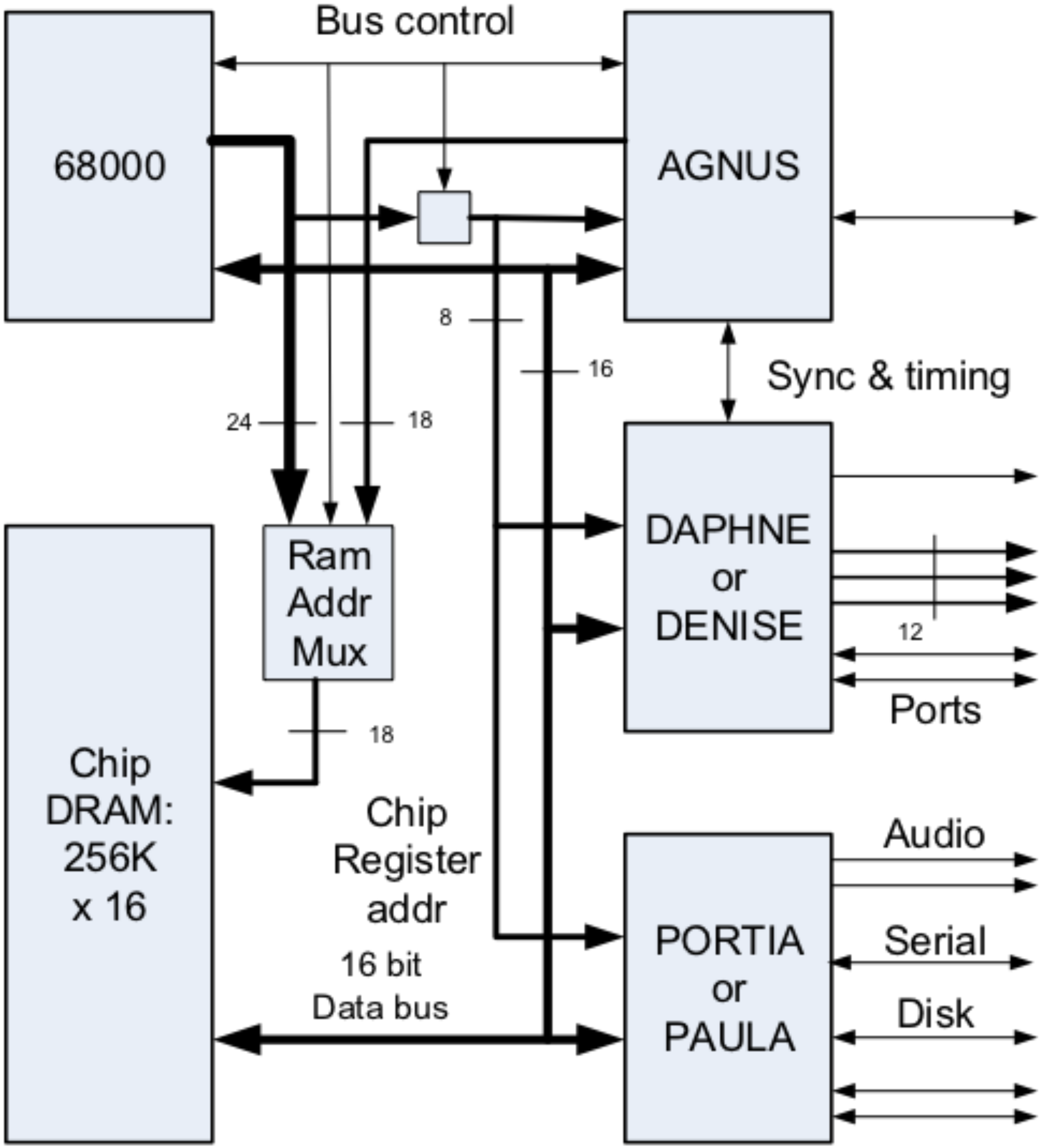
- The COPPER in the Amiga AGNUS ASIC implements a generalized Video Display List co-processor
- It can wait for a beam position, horizontal and vertical
- It can then move data directly into other IO register locations:
  - Positions, colors, initiate blitter operations, etc.

# COPPER

The Copper can move arbitrary data to any chip register.

Examples:

- Blitter
- Scrolling
- Color palette
- Sprite setup
- Audio setup
- Interrupts



# Video Bit-map Hardware

- Two resolutions: 320 or 640 across horizontally
- Capable of non-interlaced (240) and interlaced (480) operation vertically
- Bit-plane oriented: 1 bit (monochrome) up to 6 bits/pixel
- Why bit planes: we expected to be memory limited in a game console, and we wanted to be memory efficient
- Ability to use two sets of 3 bit planes, moved independently
- Support for horizontal and vertical scrolling

- Palette registers: with 4 bits each for color, chroma and luminance
- Hold-and-Modify: freeze two values, change the third for a compressed differential encoding (more colors)

# Bit-maps and RAM Usage

- For a TV, 320 horizontal and 240 vertical (non interlaced) is a full screen
- One bit plane (monochrome) is 9600 bytes
- Three bit planes (8 color) is 28800 bytes – most of the original 32KB design (game console version)
- A split system with 3 bit planes for a background and 3 bit planes for bit-blit sprite objects, would be 57600 bytes, nearly half of the original 128KB shipping design
- For high res, a 640 x 480 (interlaced) screen is 38400 bytes per bit plane
- 640 x 480 x 8 colors (3 bits) would need most of the original shipping RAM

# Bit-blitter Hardware

I first demonstrated software bit-blitting on an Apple II

We designed a system that would read in multiple bit planes, perform logical operations on them, and write the results

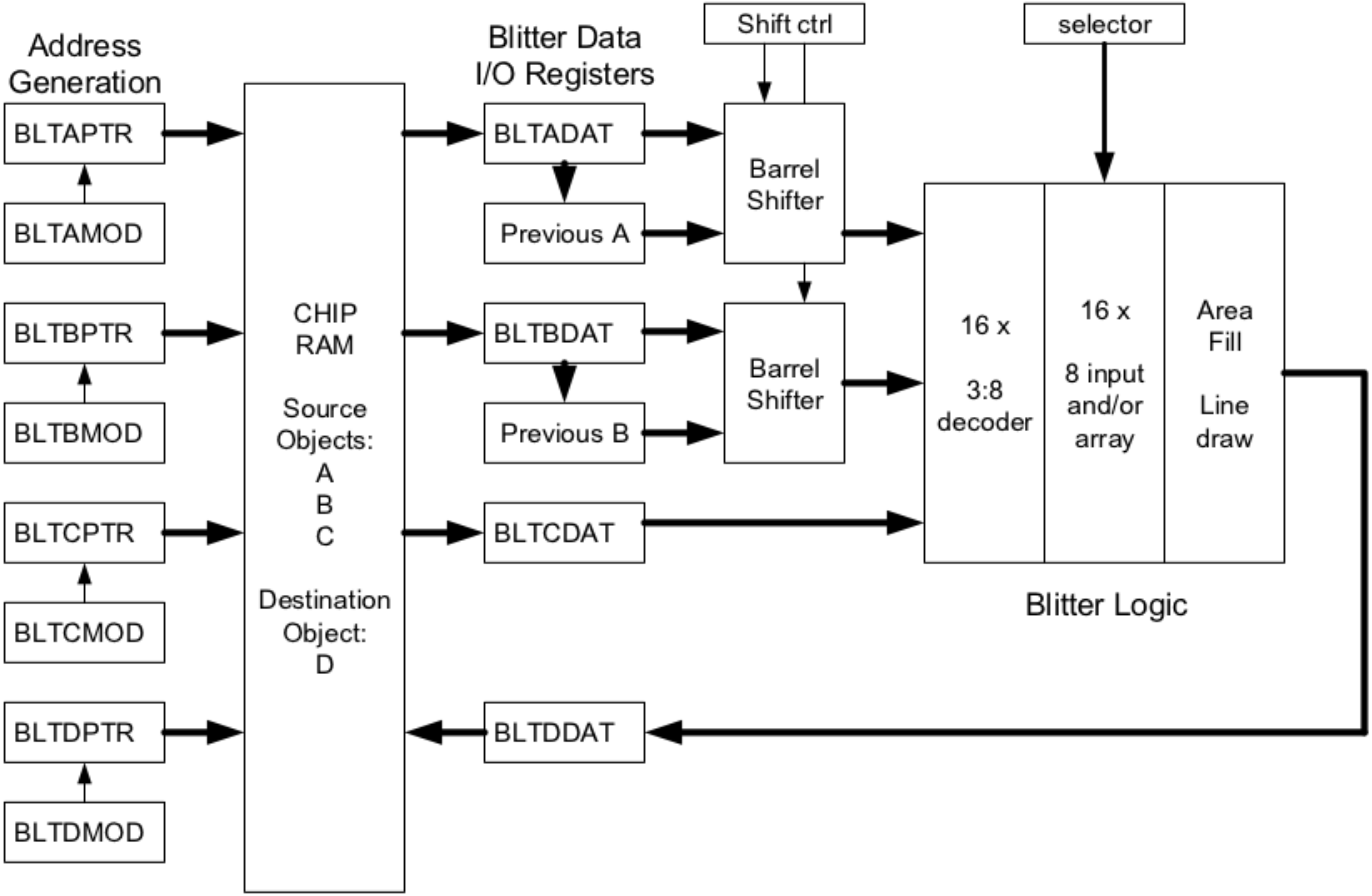
The original use case:

- Take a background image
- Take the graphics for the character or game object
- Take the outline of the sprite, and use it as a ‘cookie cutter’
- The outline would chose the object, or the background, on a bit by bit basis
- The result would be written to the composited image

Generalized logic design for Blit logic flexibility

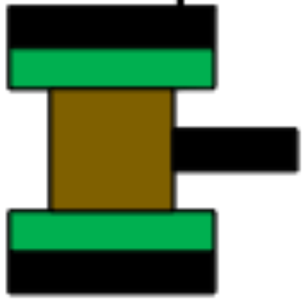
- An 8 bit mask choses which decoded minterms are OR’d into the result
- 3 input LUT (look-up-table) similar to an primitive FPGA logic cell

# Amiga Bit Blitter Diagram (Agnus)



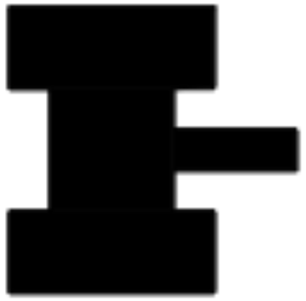
# Illustrate Bit-blitter Splicing a Sprite

Tank sprite



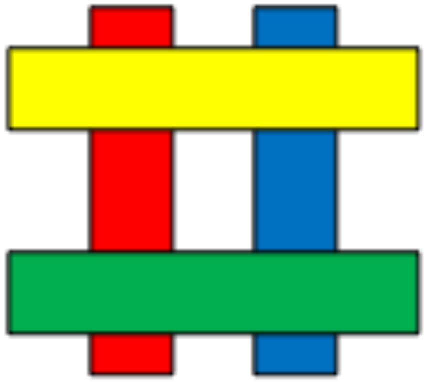
Blitter Input 1

Tank outline



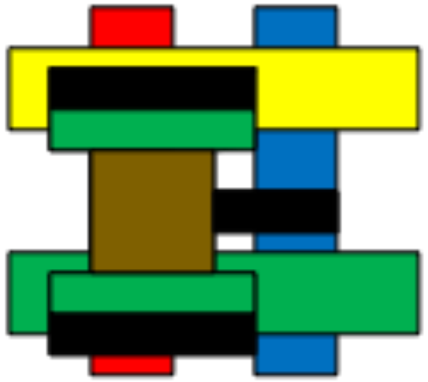
Blitter Input 2

Random background



Blitter Input 3

Blitter set:
Output =
1and2
Or
3and2not



Blitter Output

# Bit-blitter Enhancements

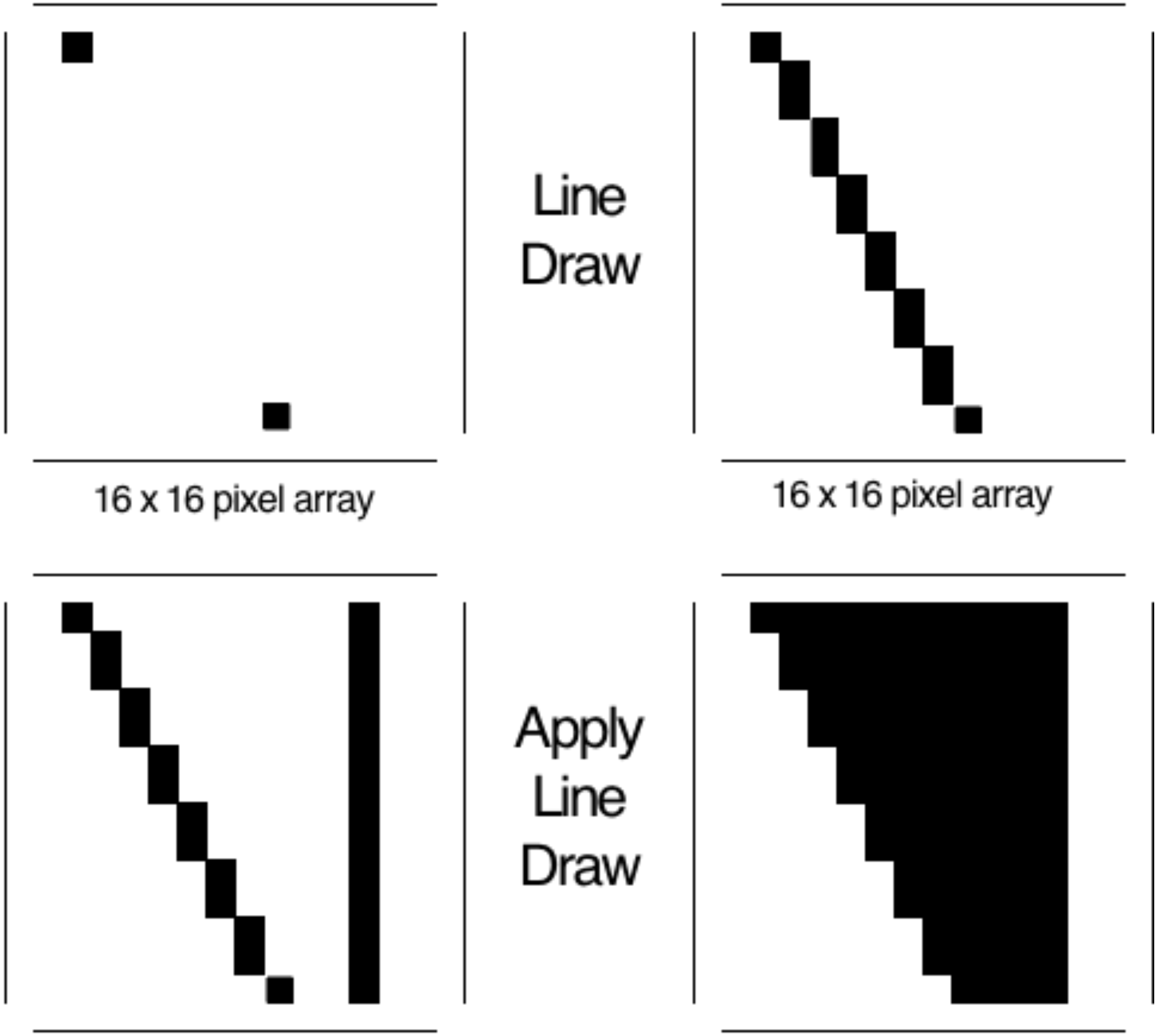
The first use case depends on an artist composing the outline of a character

The second use case: draw a line in a rectangular chunk of image memory

- The enables drawing polygons

The third use case: given a memory image with a single width drawing, do area fill on the polygons

# Line Draw & Area Fill



16 x 16 pixel array

Line Draw

16 x 16 pixel array

Add a second line

Apply Line Draw

# Sprite Engines

The Atari VCS had 5 sprite engines

- Two 8-bit objects; three 1-bit objects
- Reusable vertically in game firmware

The Atari PCS had 8 sprite engines

- Four 8-bit objects; four 2-bit objects
- Reusable vertically by hardware

The Amiga has 8 general purpose sprite engines

- 16 bits wide and 2 bits deep
- Each sprite engine executes a list
- This list mixes positioning information and image graphics

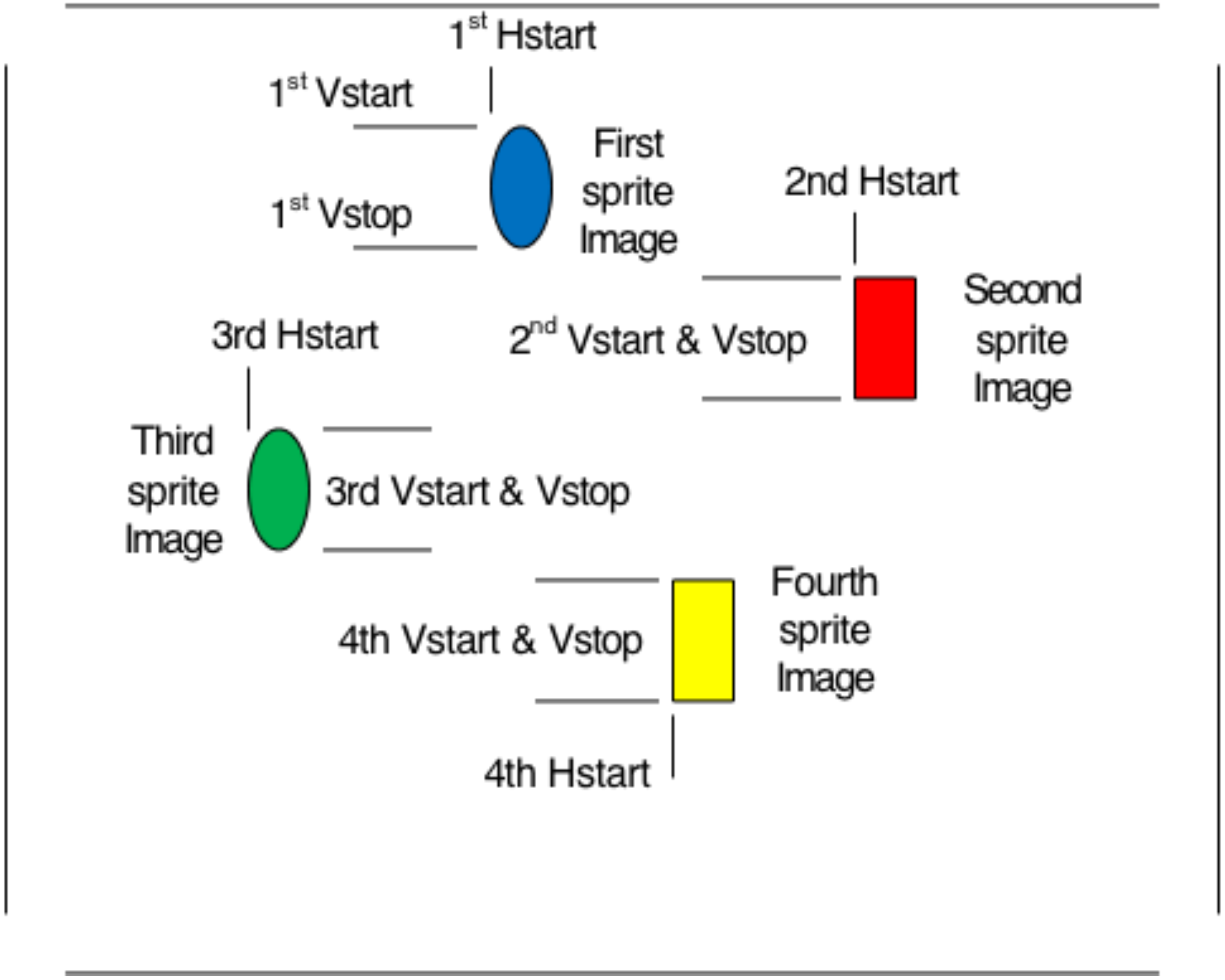
For implementing a game, these engines can generate large numbers of complex objects with low software or bit-blitter effort

- As an example: the arcade game Galaxian would be easy
- A football game with 2 dozen players, a ball and officials, would be straightforward

# Using a Single Sprite Engine

32 bit records:

- Sprite control field:
  - Vstart
  - Hstart
  - Vstop
  - Control bits
- Sprite image:
  - 16x2 lines
- Optional:
- More sprites



# Audio

# Audio Engine

There are 4 independent audio channels

Each channel can sample an 8-bit audio waveform

Each channel divides the 3.58 MHz color clock by its own 16-bit divisor to create its audio sample rate clock

Memory pointers fetches 8-bit samples (in pairs)

DMA Memory pointer starting positions & lengths

The 8-bit samples are multiplied by a volume register

PWM plus resistor ladder to do the multiplication

The 4 channels are summed and fed to the audio output

# Some Audio Details

4 independent channels

A waveform segment can be sampled and played back

Example waveforms: sine wave, triangle, or square wave
Long pseudo-random pattern for noises

As a demo, we recorded the sound of Jay’s dog barking, and controlled the pitch of the barks with the keyboard

# Finished System and History

Simplified Amiga System ASIC Architecture

When the 68000 drives the buses:

- 18 bits address for RAM read or write
- 8 bits chip register address for chip read or write
- 68000 16 bit data bus

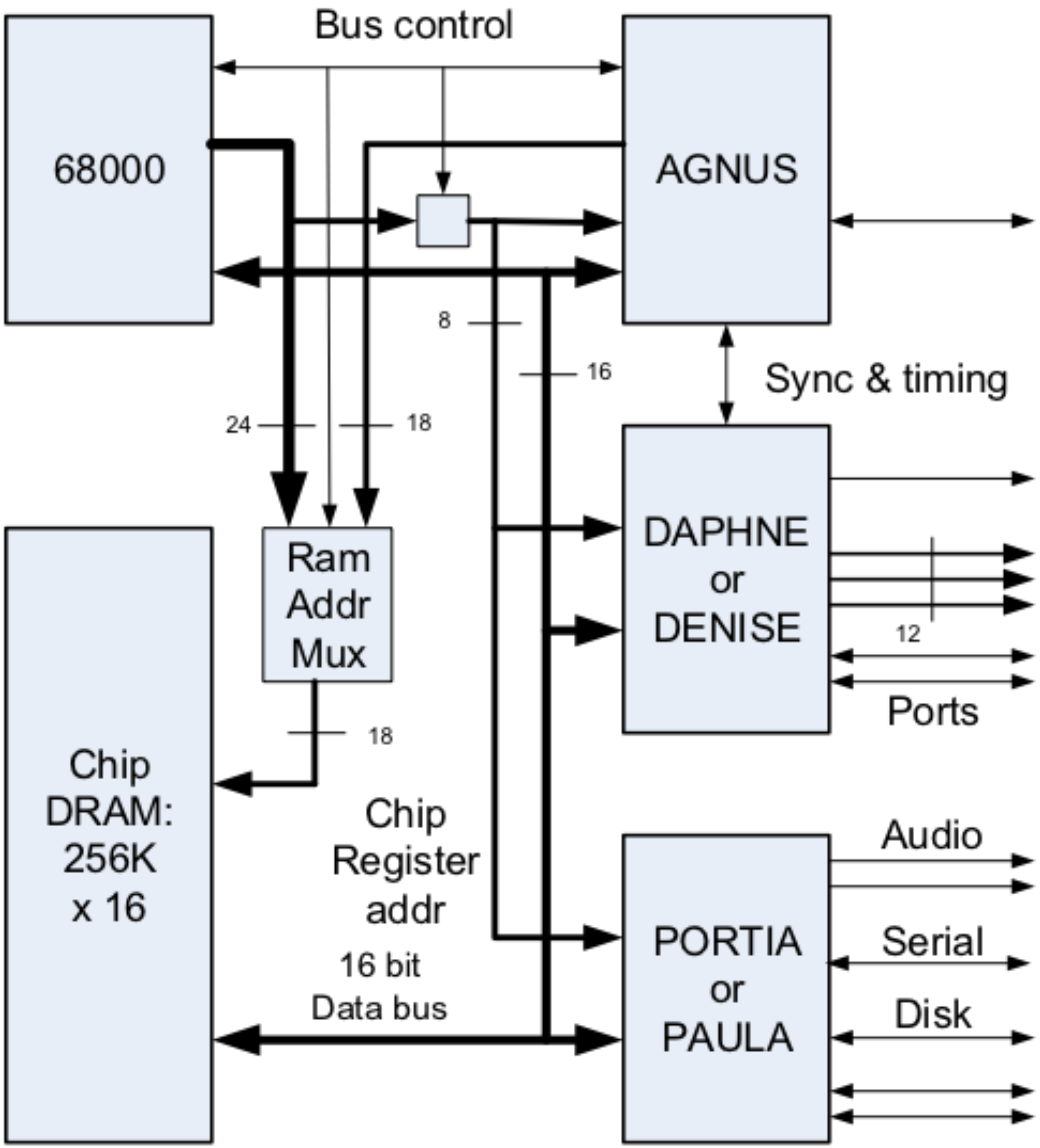
When AGNUS drives the buses:

- 18 bits address for RAM read or write
- 8 bits drive the chip register address for chip read/write
- Data flows between the RAM and chip registers on the data bus

Example AGNUS operations:

- Blitter data into AGNUS
- Bit-map data into DAPHNE
- Audio data into PORTIA

Example: Copper can drive IO ASICs



# Other Details

DMA for raw-transition-encoded Disk I/O

Serial port (including MIDI baud rate support)

3 chip partition (pinout & die size constraints)

Agnus, Daphne & Portia (later renamed)

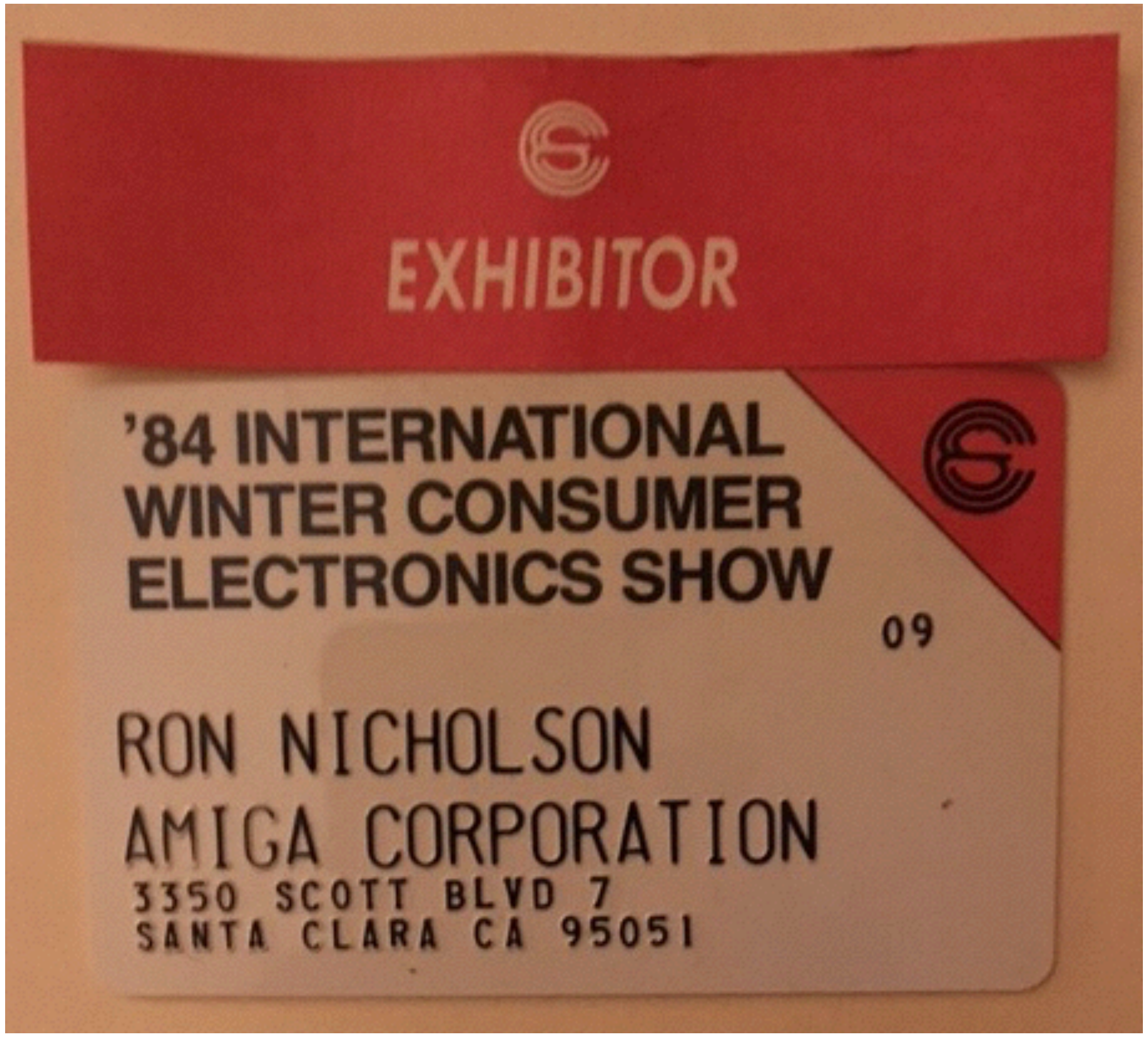
YIQ color output (later changed to RGB)

Dale Luck added Line Draw capability

# Project History

- Whiteboard block diagrams, and timing diagrams for the system, then for the 3 chip micro-architecture; create register maps
- Wire-wrapped two 68000 motherboards (only one was reliable)
- Detailed circuit design and logic design for 3 custom chips
- Hire Software Team
- Wire-wrapped 3 very large TTL chip emulators (2 boards)
- CES Trade Show
- No CAD tools!  Manual chip layout, manual logic & DRC checks
- ASIC fabrication of the 3 chips (4 micron depletion-mode NMOS)
- The first silicon all mostly worked (except for one audio bug)
- Re-positioned the game machine as a personal computer
- Software team builds multi-tasking Amiga OS
- First product: Amiga 1000

# CES Trade Show



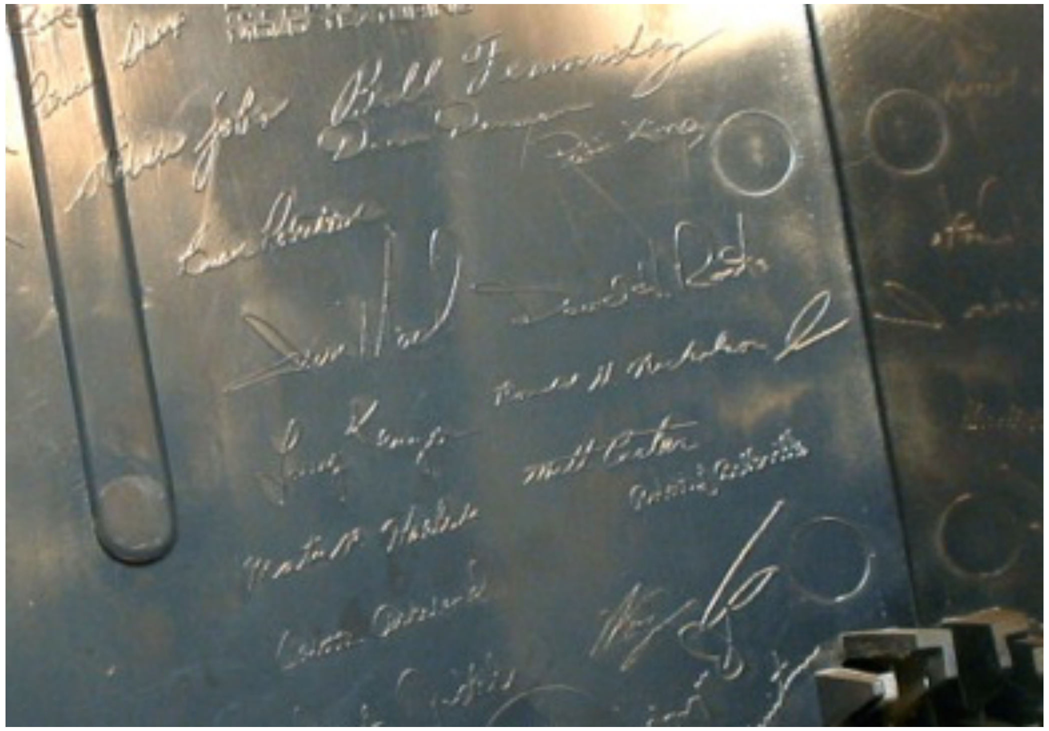
EXHIBITOR

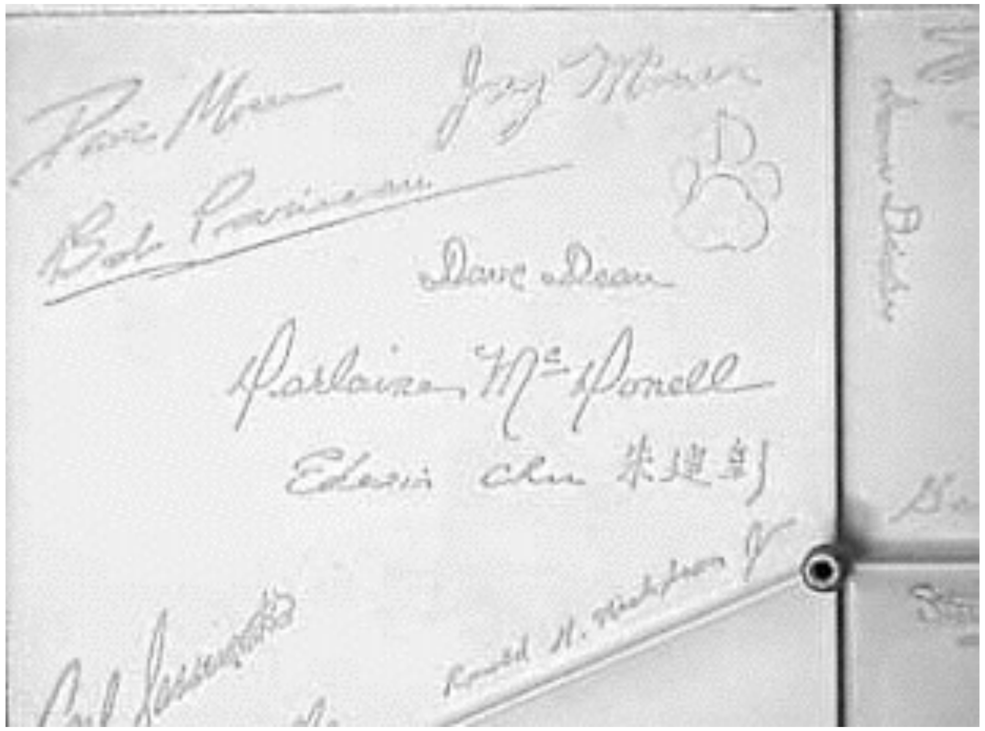
'84 INTERNATIONAL
WINTER CONSUMER
ELECTRONICS SHOW

09

RON NICHOLSON
AMIGA CORPORATION
3350 SCOTT BLVD 7
SANTA CLARA CA 95051

# Case Signatures





# Summary

Architecture and hardware design of a low-cost high-performance graphics and personal computer system with 3 full custom ASICs by an insanely small team of engineers

A creative and innovative feature set that kept the Amiga system popular among users for an unimaginably long time

# To Learn More

Amiga Hardware Manual, Commodore Business Machines, 1985

The Computer that Would not Die, Paul Wallich, IEEE Spectrum, March 2001

Commodore: A Company on the Edge, Brian Bagnall, 2010

MIT Press published a Platform Series book “The Future Was Here”, Jimmy Maher, 2012

In Memory
of
Jay Miner

Leader and
Mentor

# Contact Information

Joe Decuir
Issaquah, WA USA
email: jdecuir@ieee.org

Ron Nicholson
Santa Clara, CA USA
Email: rhn@nicholson.com
@hotpaw