DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Rev 7-19-23

Tech History

Part 2

Chips (IC's & MPU's)

by

Dr Jeff Drobman

Dr Jeff Software

Index

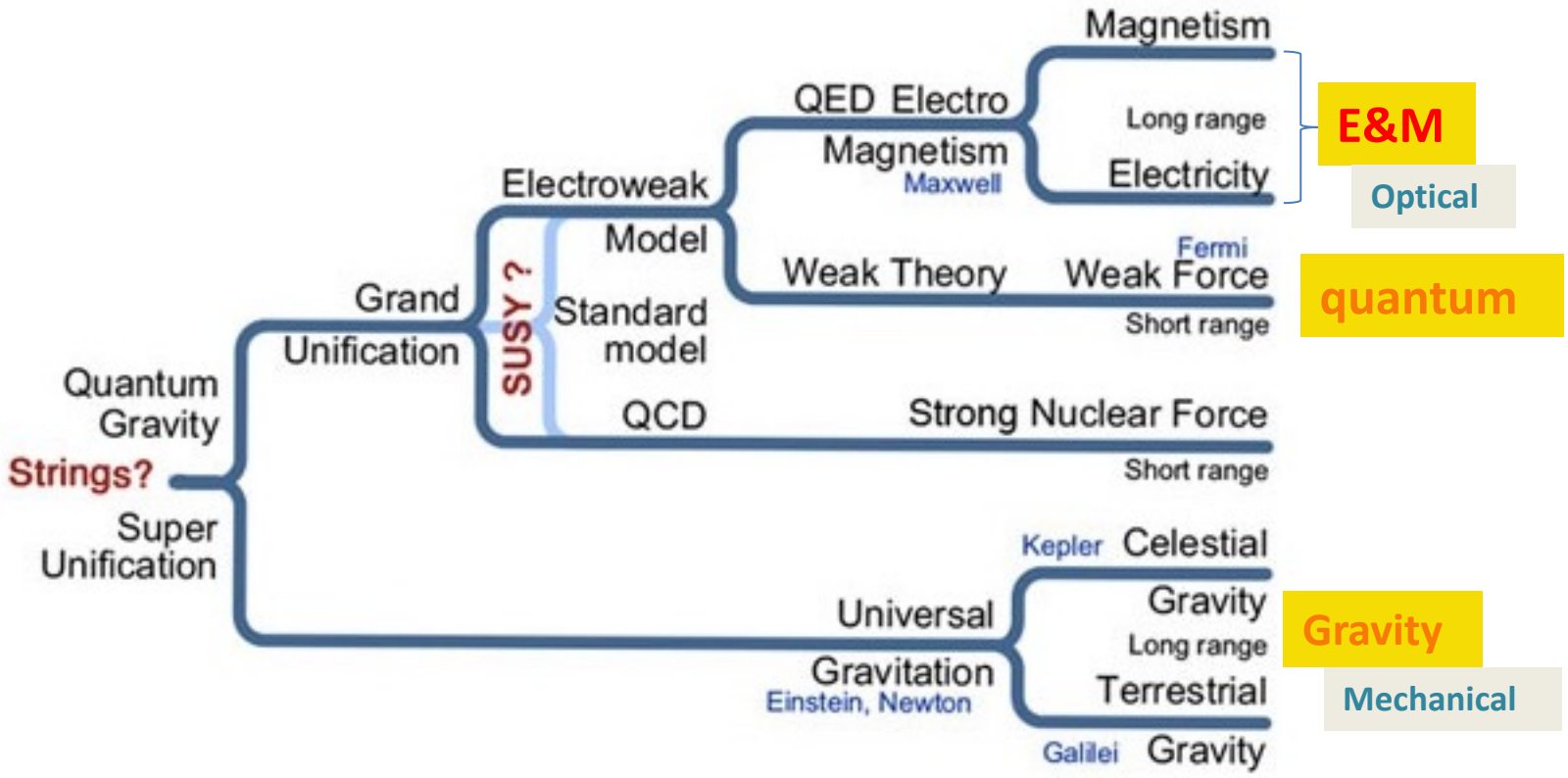
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Part 2: **Chips**

Physics of Computers

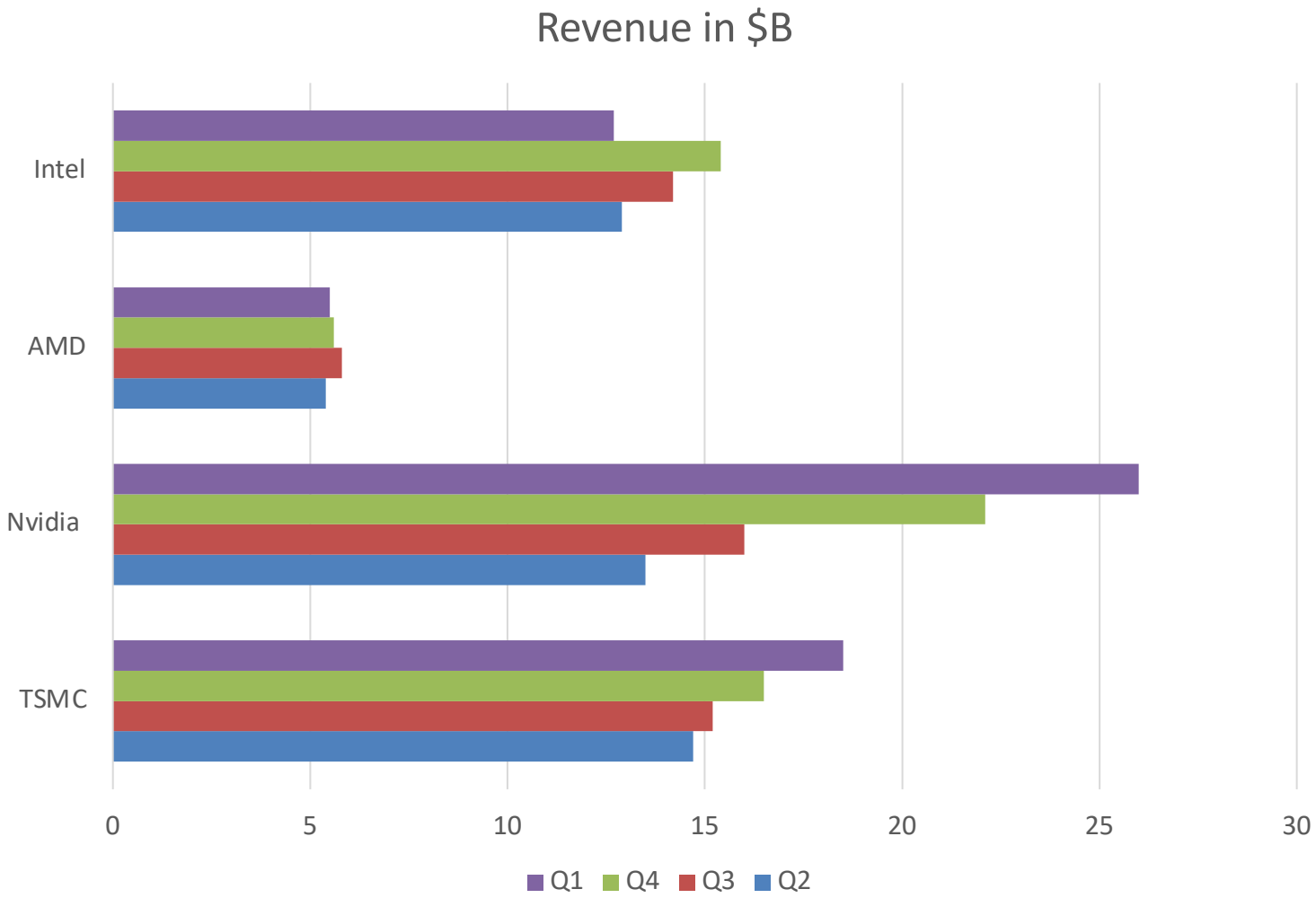
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

4 Forces in Nature



Chips 2Q23-1Q24

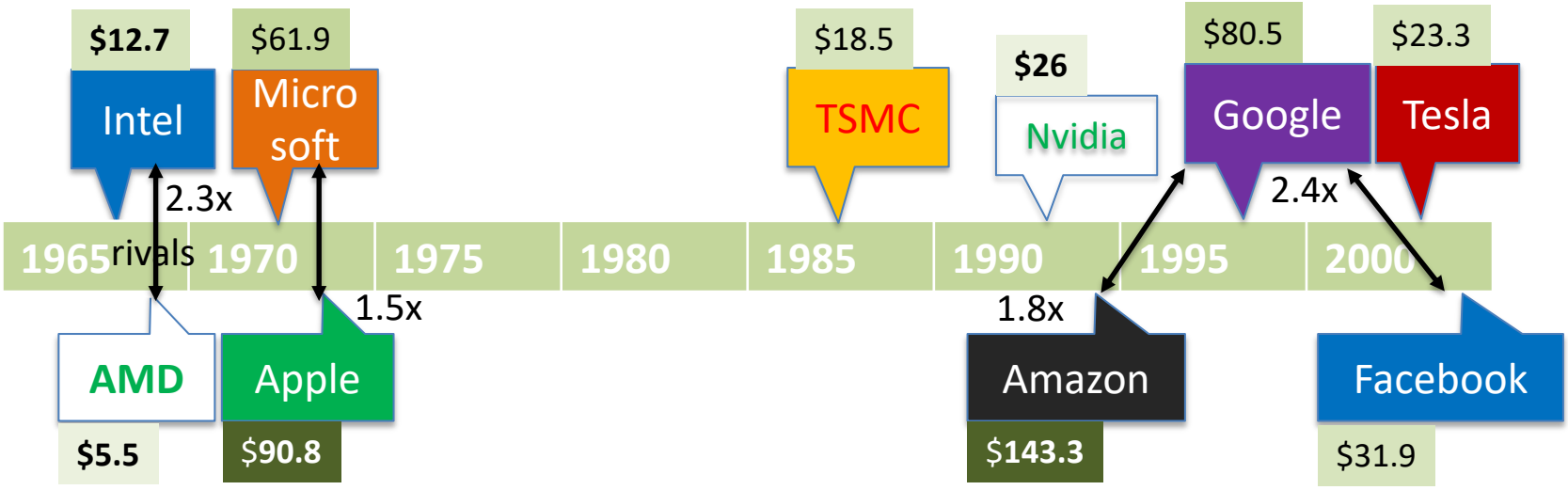
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



Tech Titan Timeline

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Qtr Revenue in $B | Current Perspective | **Q1** 20**24**



2Q23

- Other *Industrials*
  - **GM** $40.0 → 2x Tesla
  - **Ford** $39.1
  - **IBM** $16.7 → ~Intel
  - QCOM $9.3
  - TI $4.2
  - NXPI $3.3

2Q23

- Other *Services*
  - **Netflix** $8.2
  - Visa $8.1
  - **PayPal** $7.4

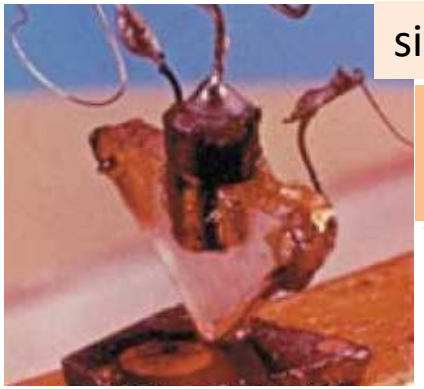
Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Transistors

The Transistor

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



size = ~1 inch

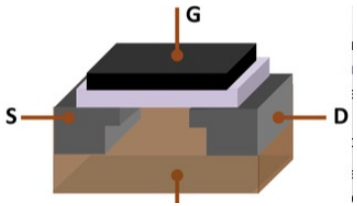
1947 ushered in the era of ***Microelectronics***

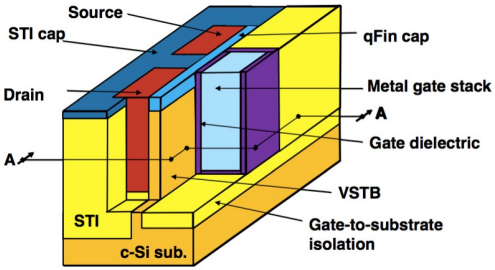
A **transistor** is a semiconductor device used to amplify or switch electronic signals and electrical power. It is composed of semiconductor material usually with at least three terminals for connection to an external circuit. A voltage or current applied to one pair of the transistor's termina

- 1947- Bipolar point/junction
- 1959- Planar bipolar [10]*
- 1964- MOS (P-channel) [100]
- 1972- MOS (N-channel) [1,000]
- 1978- CMOS [4,000]
- 1990- sub-micron [10,000]
- 2000- 100 nm [100,000]
- 2011- FinFET [1,000,000]
- 2022- 5nm [50,000,000,000]

*no. of transistors

Transistors have been shrunk every 2 years according to ***Moore's Law***





- size = 10 nm = 4×10^{-7} inches
- yields → ~1M devices per cm^2

Viewing Transistors

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Quora

How are billions of transistors compressed into a single chip? Can a transistor in the chip be seen with a microscope?

Jeff Drobman
Works at Dr Jeff Software · Just now ·

once a chip is complete, only the top few layers of metal interconnect are visible (unless etched away). a single transistor is way too small to be readily identified, as they are now as small as about 25–40 nm in overall size. note that a “5nm” node means that only the channel length is that small. the overall transistor size, and pitch, is more like 25–40 nm.

Transistor Atoms

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Quora

Al Kordesch, Semiconductor Device Modeling

Answered Feb 1, 2019

How many atoms are in a typical transistor in a chip?

Short Answer: 49,000 atoms!

Apple's iPhone XS uses 7 nanometer transistors. So let's estimate how many atoms are in one of them. Excluding the connecting wires and other parts, I'm just going to calculate the size of the active part, the "channel" under the gate. The volume of the channel is about (7 nm long) x (7 nm deep) x (20 nm wide). The atomic density of silicon is 5E+28 atoms per cubic meter. So let's go!

Number of atoms n = volume x density

n = (980E-27) x (5E+28) = 49,000 atoms.

Silicon atomic radius = .111nm → **4.5** atoms/nm

Cubic: 4.5^3 = **91** atoms/cu nm

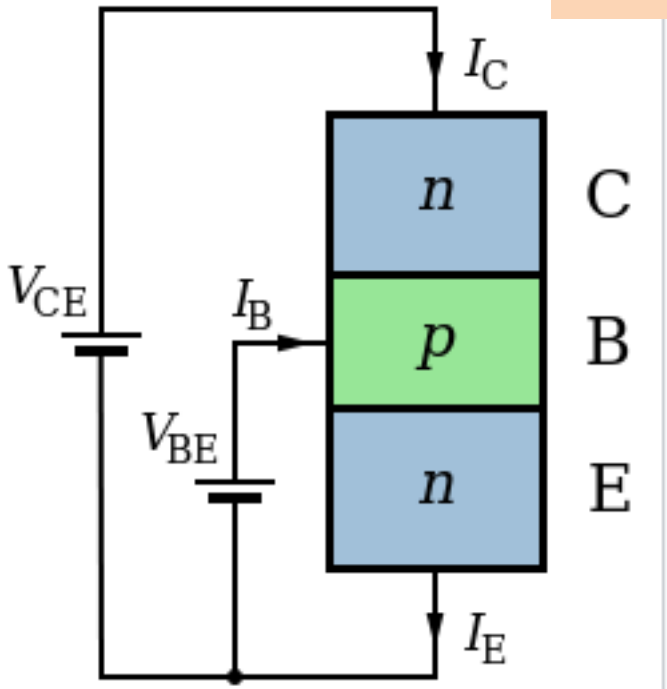
Channel volume @5n: 5x5x18 = **450** cu nm

Cubic: 91*450 = **40,950** atoms/channel

Bipolar Transistors

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

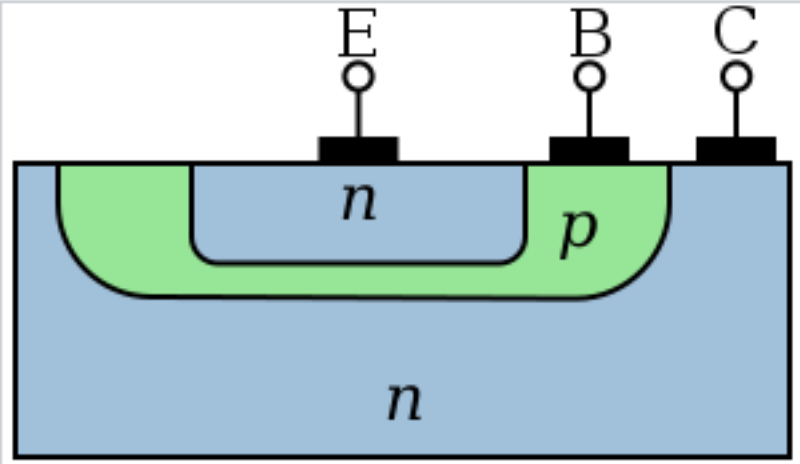
BJT



Structure and use of NPN transistor. Arrow according to schematic.

PLANAR

Structure [edit]

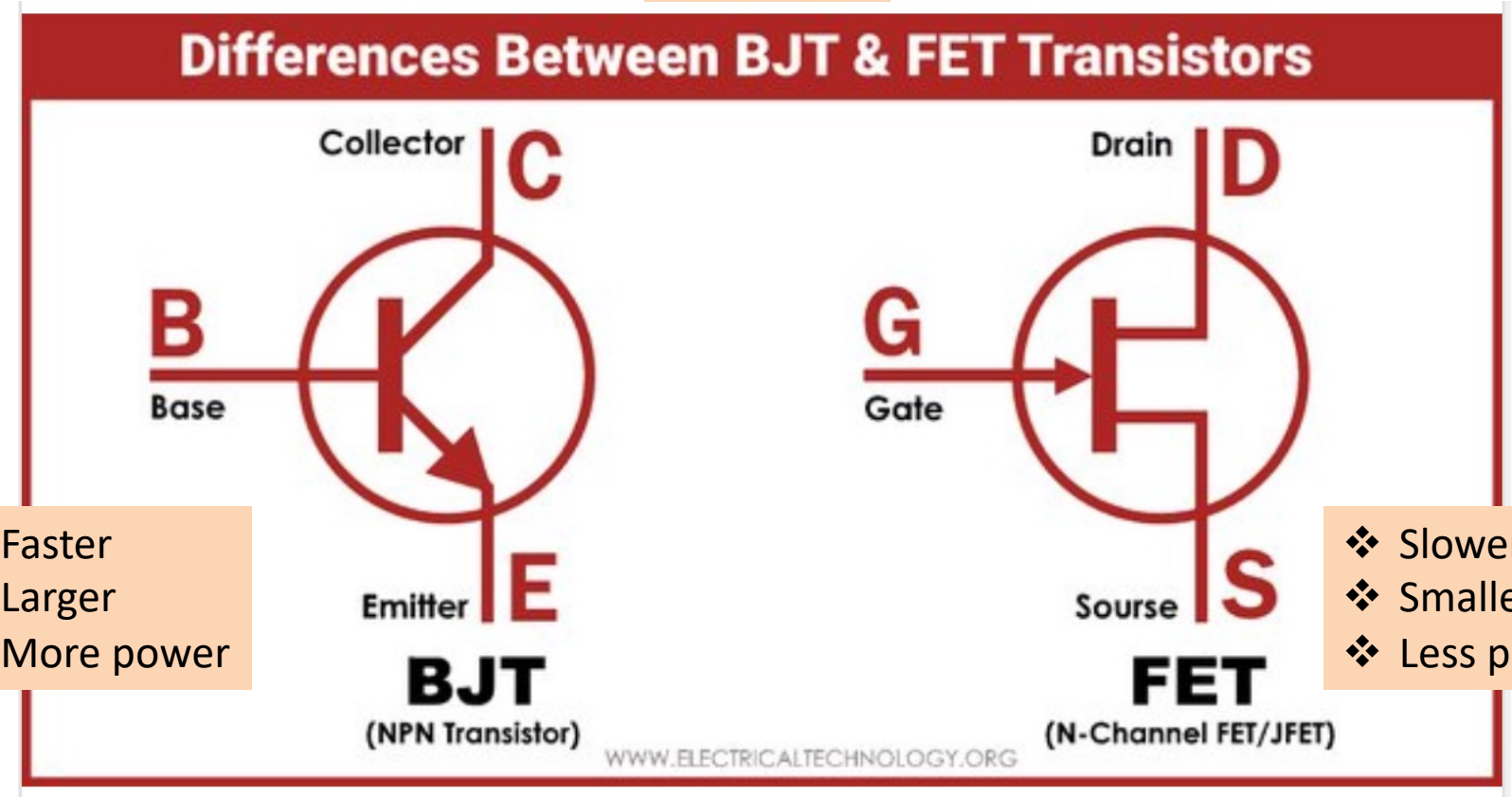


Simplified cross section of a planar *NPN* bipolar junction transistor

Bipolar/MOSFET Transistors

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Schematics



- Faster
- Larger
- More power

- Slower
- Smaller
- Less power

- Current flows opposite electrons (C->E, S->D)
- B, G are inputs (H/L)
- B, G voltages turn transistor ON/OFF
- Outputs (not shown) are tied to C, D

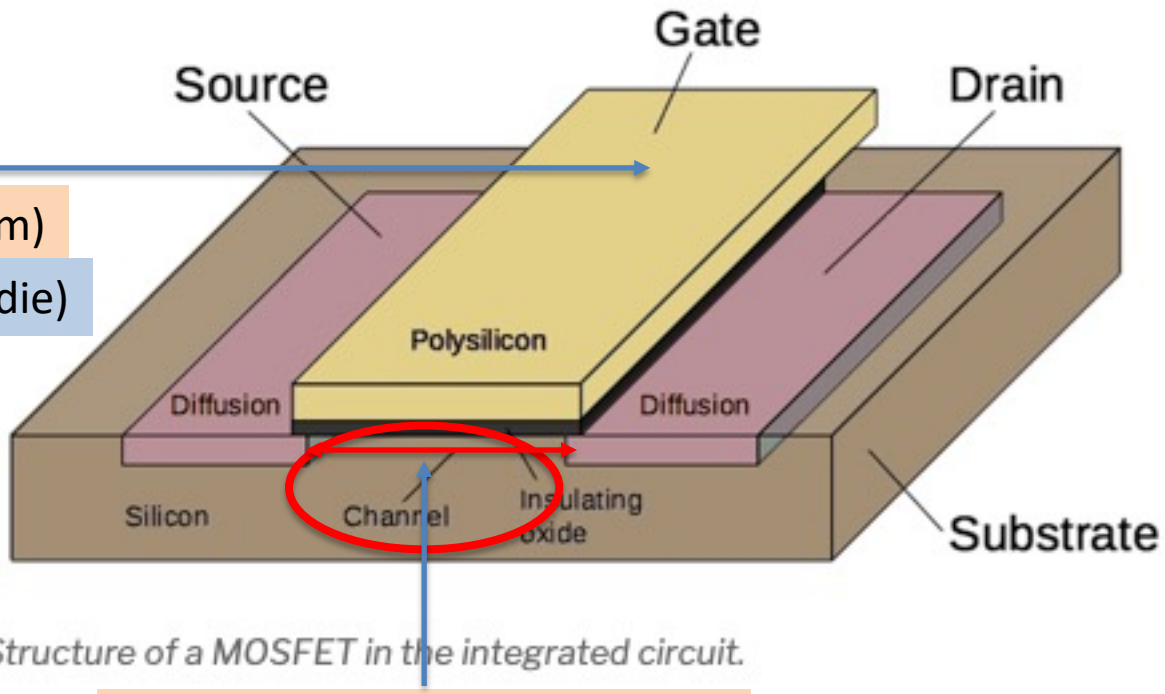
MOS Transistor

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

WikiSemi

$V_g = 5 \rightarrow 3.3 \rightarrow 1.8$ V

Near Gate



Structure of a MOSFET in the integrated circuit.

Pitch (e.g.14nm)

Density (xtor/die)

Smallest feature size (e.g. 5nm)

Performance (F_t)

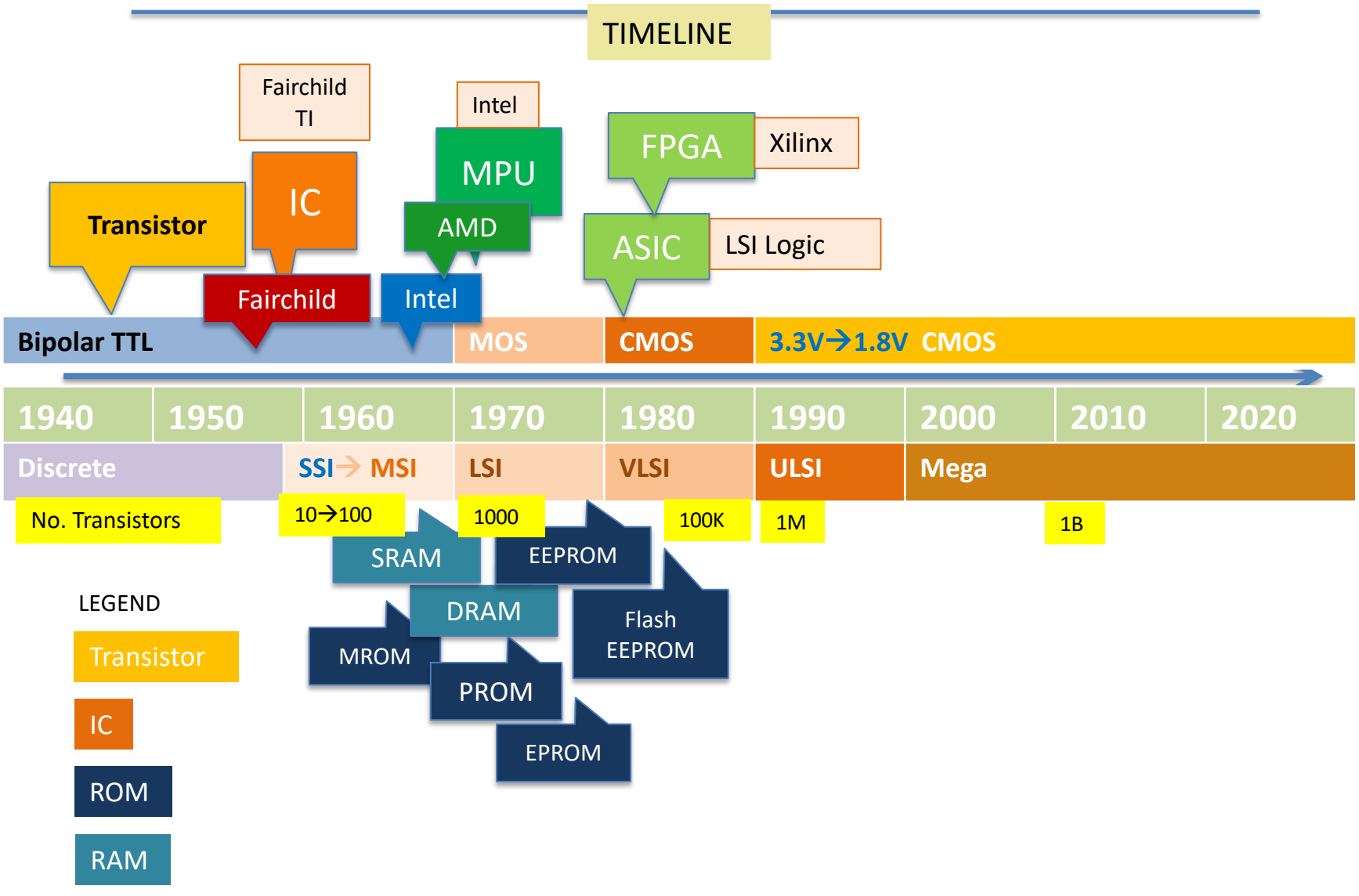
Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Chip History

Levels of Integration

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



© 2022 Dr Jeff Drobman – drjeff@drjeffsoftware.com

IC Process & Interface

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

TTL compatible — **5V → 3V**

❖ *Bipolar*

5V

➢ RTL→DTL→TTL → *Schottky* TTL→ *LS* TTL

1960	1965	1970	1975	1980	1985	1990

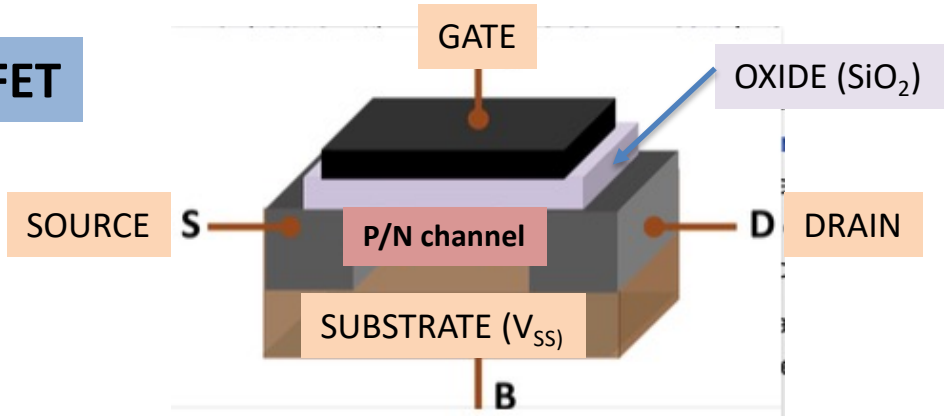
❖ *MOS*

➢ PMOS→NMOS → **CMOS** → CMOS (TTL I/O)→ 3.3V

BiCMOS

5V → 3V

MOSFET



The **metal–oxide–semiconductor field-effect transistor**, also known as the **metal–oxide–silicon transistor**, is a type of field-effect transistor that is fabricated by the controlled oxidation of a semiconductor, typically silicon. It has an insulated gate, whose voltage determines the conducti

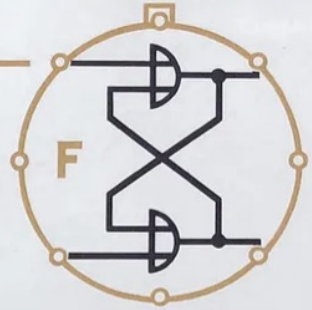
1st Flip-Flop Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Fairchild 1961

ANNOUNCING THE FIRST OF A FAMILY

THE MICROLOGIC FLIP-FLOP



HIGH SPEED
LOW POWER
LOW COST

COMPATIBLE
MINIATURE
CONVENIENT

NOW AVAILABLE

The first element of the micrologic family of digital functional blocks is now available. The flip-flop, like other members of the family, operates over the full military environment at bit rates above 1 mc.

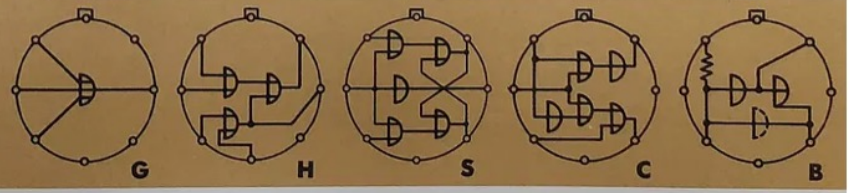
These integrated semiconductor logic elements are not laboratory curiosities; they are designed to be:

RELIABLE — The new order of reliability inherent in the Fairchild planar process is complemented by the use of deposited metallic film intraconnections.

USEFUL — The six elements of the micrologic family are in themselves sufficient to efficiently build the complete logic section of a digital computer or control system. The (TO-5) elements are designed for use with conventional printed circuit boards, the (TO-18) packages for use with welded-wire interconnections.

ECONOMICAL — Low component cost is achieved by use of the batch diffusion process. The logic-function capability and the pin layout of each element are such that the time required for logic design and circuit layout is drastically reduced.

ADDITIONAL TYPES TO COME

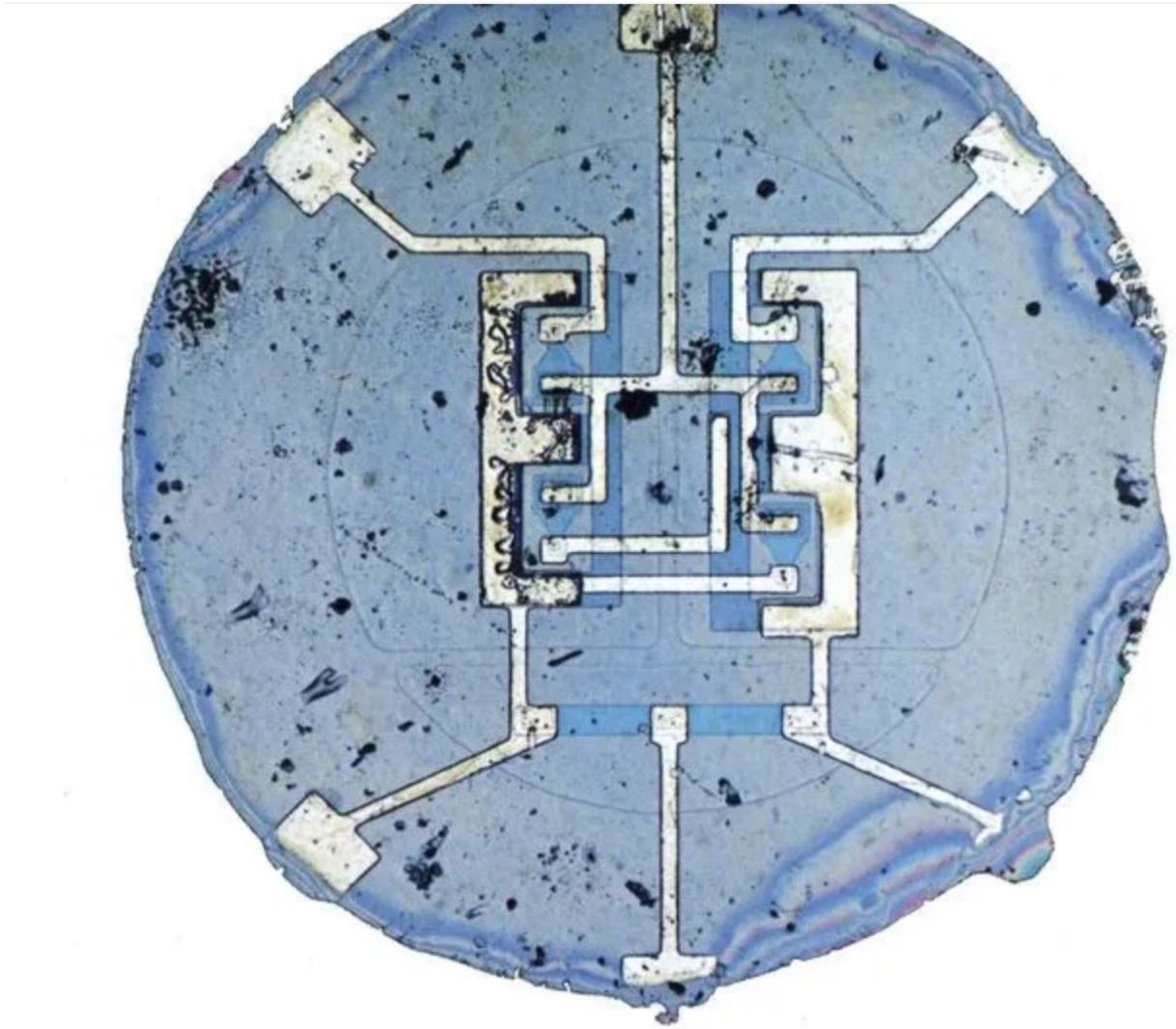


Fairchild Semiconductor 1961 Micrologic Family announcement brochure

1st Flip-Flop Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Fairchild 1961



4T

The first Fairchild integrated circuit contained four transistors. Photo: Fairchild Semiconductor

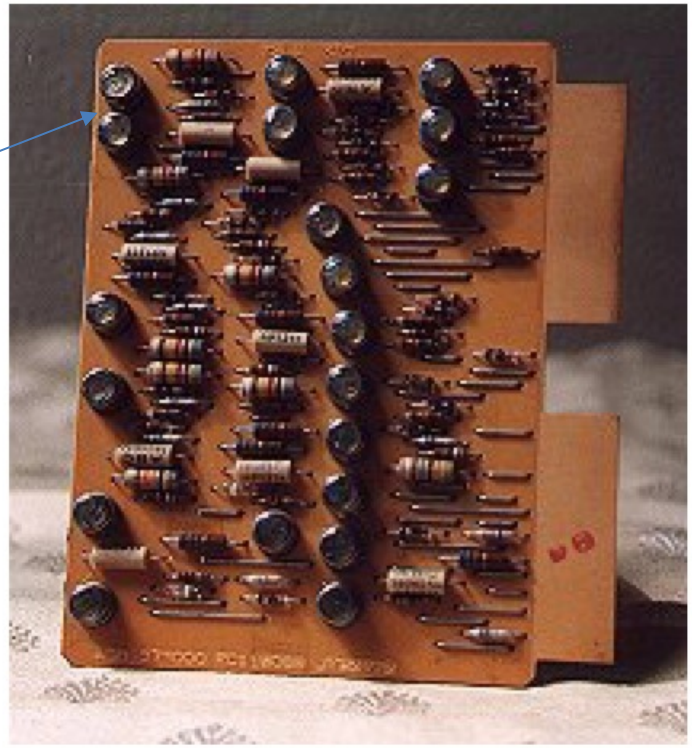
Old Computer Tech

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Discretes R/T

This is a circuit board from an IBM 7040, built in 1963.

Discrete Transistors



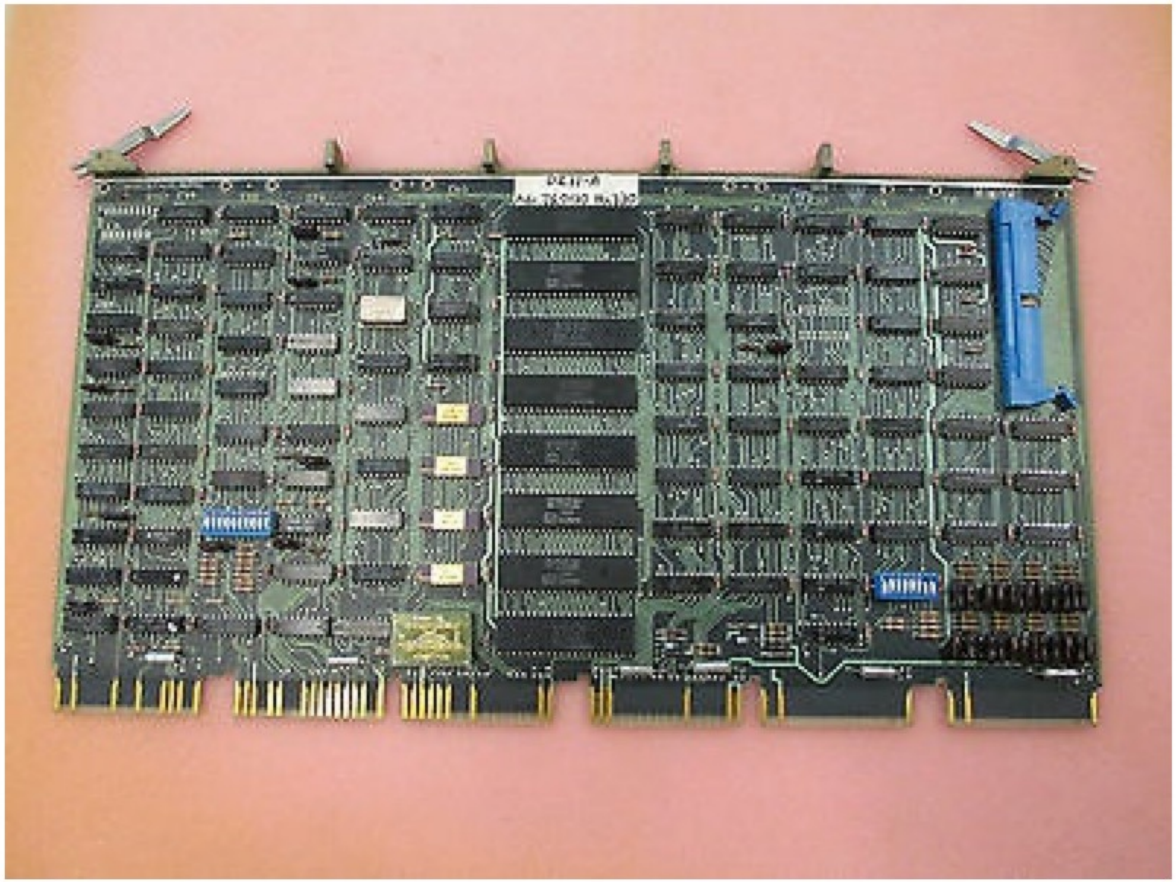
There are no small components there. Those horizontal cylinders are maybe half an inch (12mm) long.

Old Computer Tech

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

MSI/LSI IC's

This is a serial interface board for a DEC PDP-11



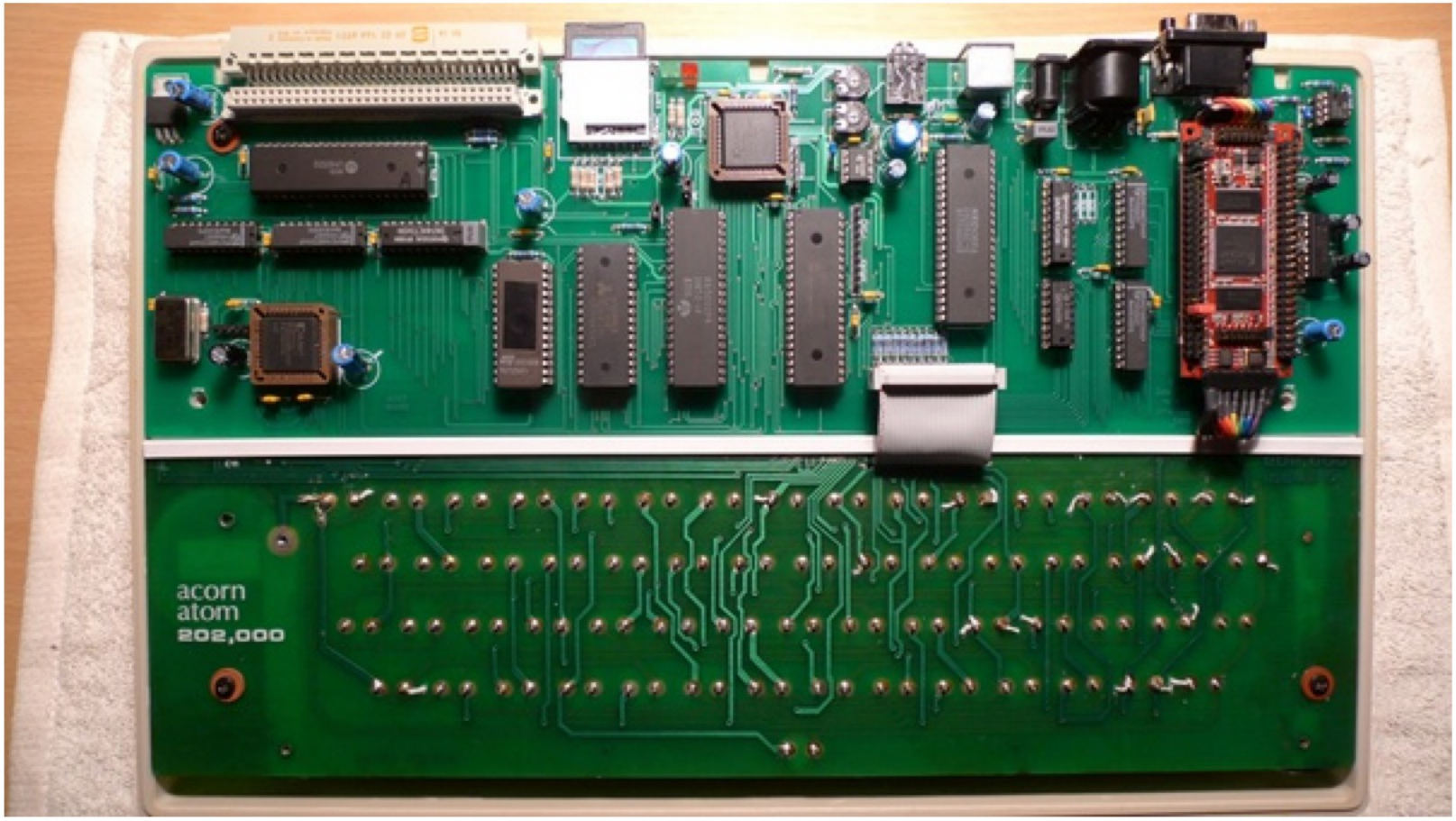
The integrated circuits were very simple with only a few dozen transistors in, so they didn't need very precise machines to make them. The boards could still be assembled by hand.

Old Computer Tech

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

LSI IC's

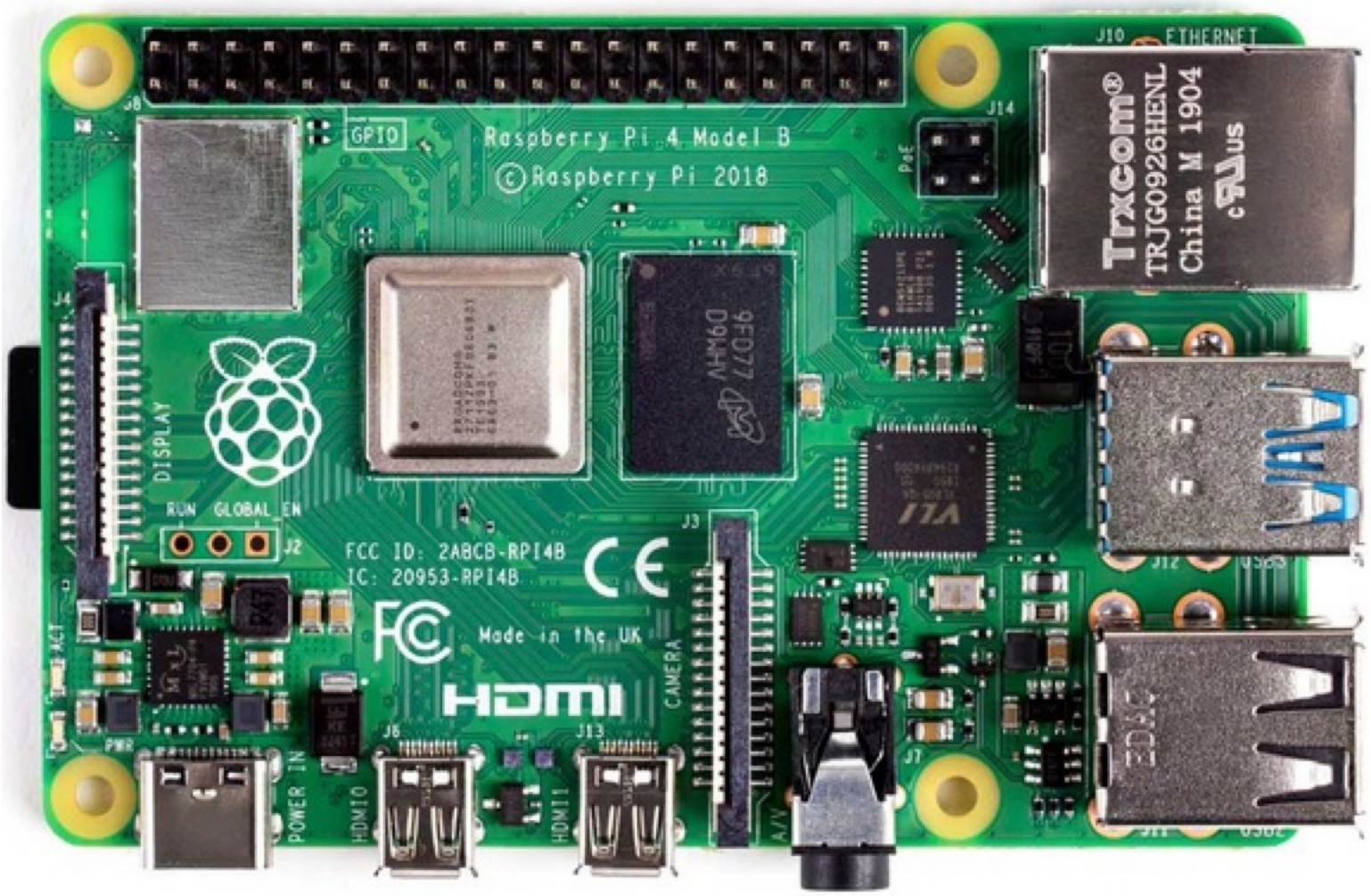
This is the second computer I assembled, the Acorn Atom



New Computer Tech

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

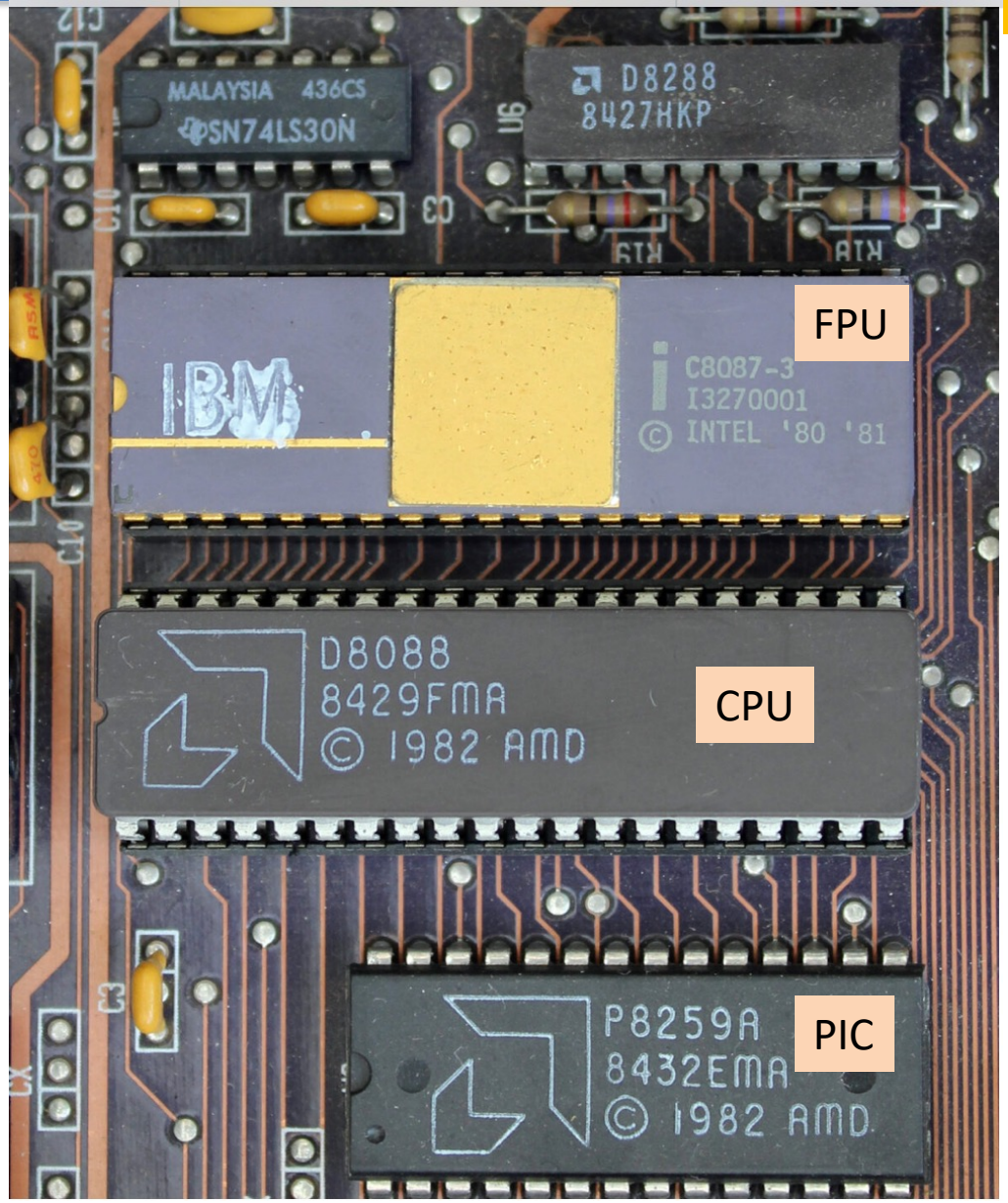
VLSI IC's



IBM PC Chips

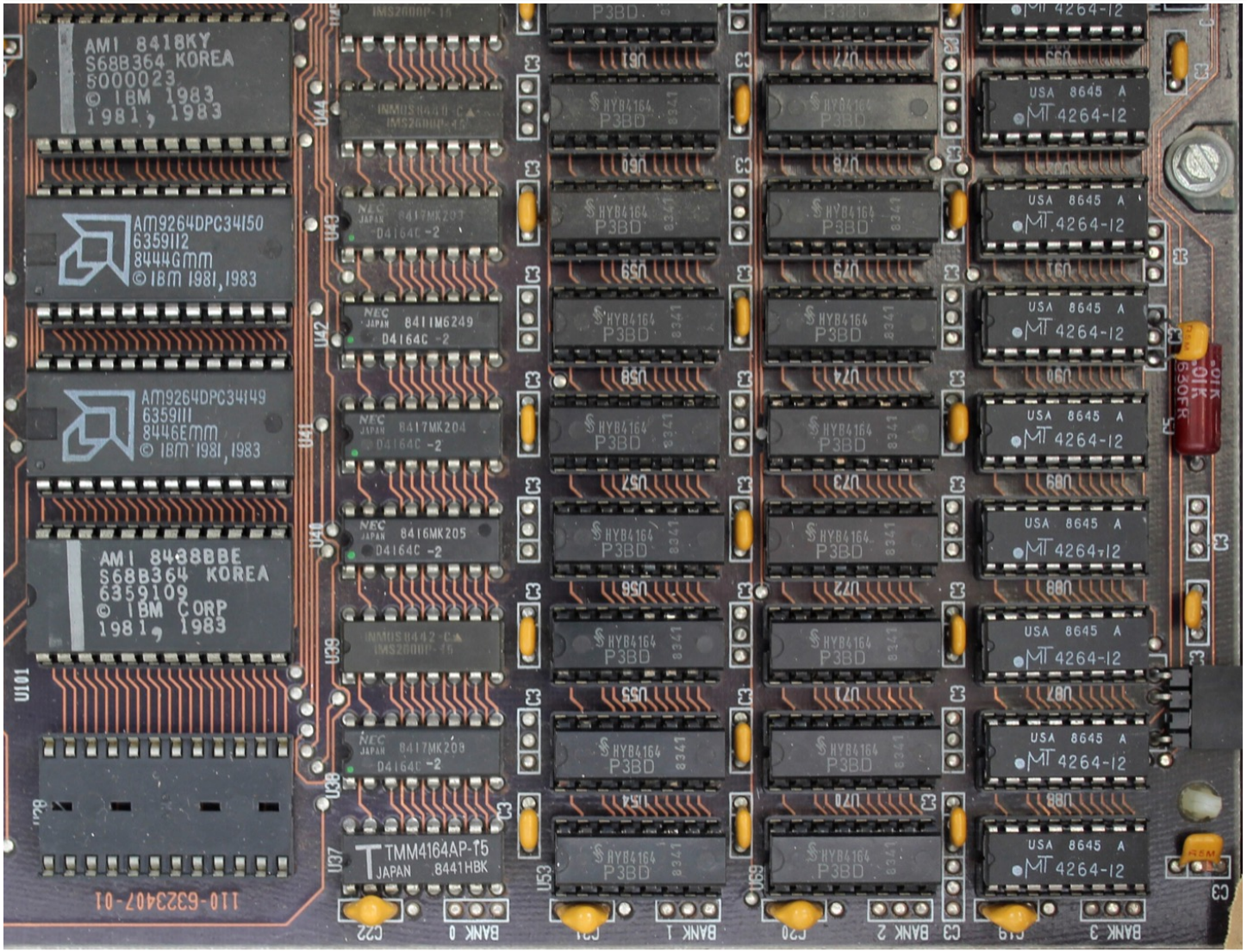
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

1981



IBM PC – Large Motherboard

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



The memory is organised in four banks in the bottom right corner of the motherboard – in this case there are four 64KB banks, adding up to a total of 256KB

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

1947-68

Early Semiconductors & IC's

MILESTONES

trans
- 1947 ❖ Transistor invented (Bell Labs' Bardeen, Brattain, **Shockley**) - point contact form
- 1951 ❖ Bipolar junction transistor (BJT) invented by Wm Shockley
- 1956 ❖ **Shockley Semiconductor Laboratory** founded as a division of Beckman Instruments
  - ➢ Shockley hires his PhD students **Robert Noyce, Gordon Moore**, et al.

Fairchild
- 1957 ❖ "Traitorous 8" leave Shockley Labs, found **Fairchild Semiconductor**
- Mar ❖ TI– **Jack Kilby** tests the world's first integrated circuit (Sgl-transistor oscillator on germanium)
- 1959 ❖ **Jean Hoerni** of Fairchild demos his "planar process" (world first)
  - ➢ **Bob Noyce** documents a method for building ICs using that planar process

IC
- May 1960 ❖ Fairchild group makes first IC
  - ➢ [Courts and the tech community decided to give equal rights to the invention of the IC to both Kilby and Noyce]

MOS
- 1963 ❖ **MOS** (linear) invented: first MOSFET amplifier demonstrated
- ❖ standard logic families are introduced using DTL and TTL structures
- ❖ **CMOS** process was invented by Fairchild Semiconductor in a 1963 paper and patent
- 1964 ❖ **MOS** (digital)– 1st products released by General Microelectronics for a calculator chipset
- ❖ **Linear** IC's– 1st analog ICs introduced by Fairchild Semiconductor
- Apr 1965 ❖ **Moore's Law** born - Gordon Moore publishes his first version
- ❖ **CMOS**– 1st parts by RCA
- ❖ **ROM**– 1st Semiconductor
- ❖ **DIP** packages

RAM
- 1966 ❖ **RAM**-- Bipolar RAMs (SRAM) introduced
- ❖ **DRAM–** IBM conceives DRAM cell (1T, 1C)
- 1968 ❖ **CMOS SRAM–** 1st parts by RCA

1968-2011

Semiconductors & IC's

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

MILESTONES

	1968	❖ **Intel** founded
	1969	❖ **AMD** founded
MPU	1971	❖ **Microprocessor** & **RAM** in **MOS** invented by Intel (i4004/i8008, i1101/3)
	1972	❖ Ion Implantation (replaces chemical diffusion) Mfg -- process
	1974	❖ Digital **Watch** IC invented
	1978	❖ **Wafer stepper** invented for fabs Mfg -- process
	1979	❖ **IDT** founded (**CMOS**) Mfg -- process
ASIC	1981	❖ **LSI Logic** founded (**ASIC**)
FPGA	1984	❖ **Xilinx** makes 1st **FPGA**, **MIPS** founded as early ***RISC*** pioneer (licenses LSI & IDT)
	1985	❖ **ARM** founded as Acorn RISC Machines
Flash	1987	❖ **Toshiba** intro's **Flash** EEPROM, **TSMC** founded (foundry) Mfg -- process
GPU	1998	❖ **Nvidia** founded (1993) – 1st **GPU's** (1998)
	2002	❖ Intel goes to **300mm** (12in) wafers Mfg -- process
	2009	❖ AMD spins off fabs to **Global Foundries** (owner Abu Dhabi) Mfg -- process
	2011	❖ Intel **FinFET** Mfg -- process
Dec	2019	❖ Intel intro's 1st **QC** chip ("Horse Ridge")
	2020	❖ ARM intro's "backside power" process Mfg -- process

Silicon Valley

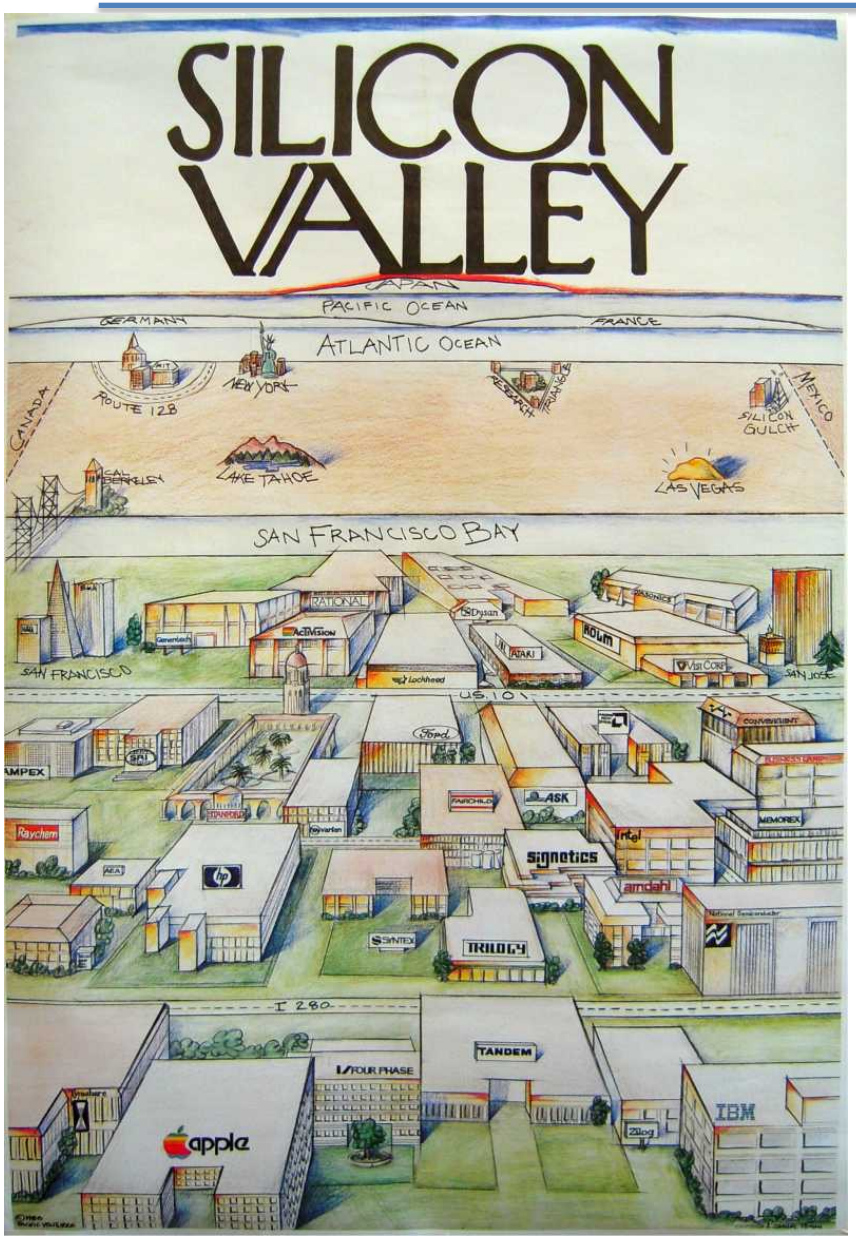
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Portion of Silicon Valley map. drawn by Maryanne Regal Hoburg (1982). Courtesy: The David Rumsey Map Center, Stanford University Library

© 2022 Dr Jeff Drobman – drjeff@drjeffsoftware.com

Silicon Valley

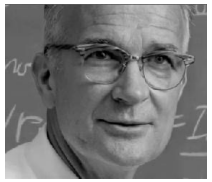
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



Founding Fathers

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

NEW
&
OLD



Missing:
- ❖ IC fathers
- ❖ Internet fathers

CHM Computer History Museum

CHM on Shockley

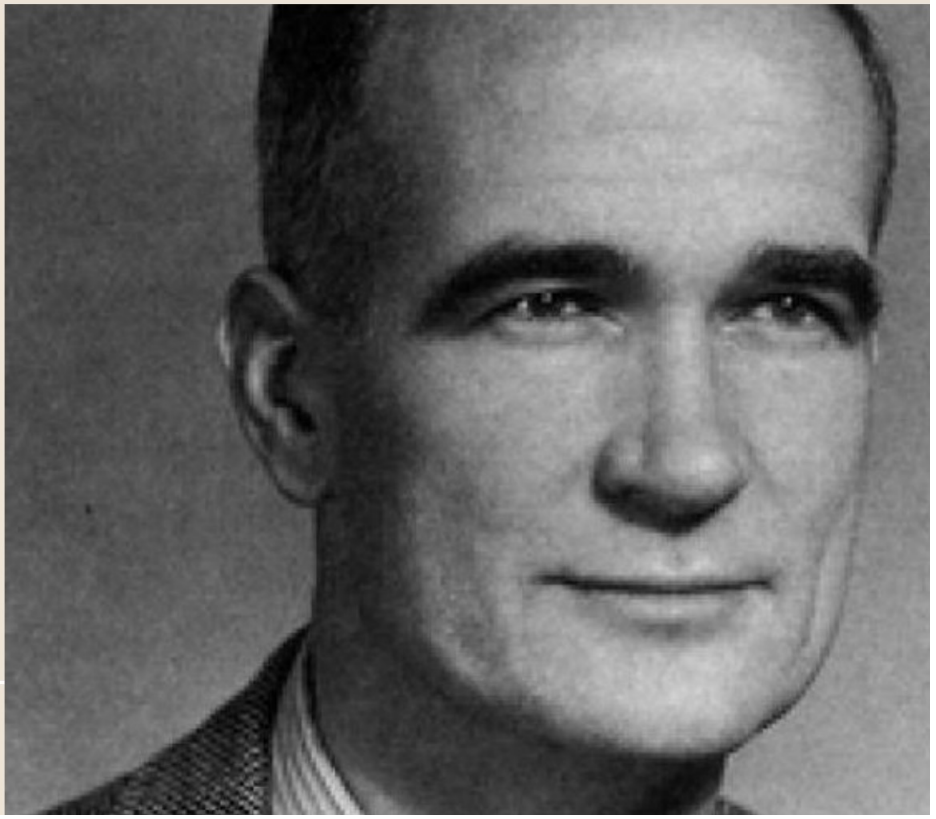
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

https://computerhistory.org/blog/beckman-shockley-and-the-60th-anniversary-of-the-birth-of-silicon-valley

CHM BLOG CURATORIAL INSIGHTS , REMARKABLE PEOPLE

BECKMAN, SHOCKLEY AND THE 60TH ANNIVERSARY OF THE BIRTH OF SILICON VALLEY

By David Laws | February 10, 2016



CHM Computer History Museum

CHM on Shockley

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

None would have the same lasting impact on the fortunes of the future Silicon Valley and beyond as Dr. Arnold Beckman's disclosure of an agreement signed the previous day for "the establishment in the Stanford community of the Shockley Semiconductor Laboratory to develop and produce transistors and other semiconductor devices."

ABOUT THE AUTHOR

David A. Laws [AMD 1975-1986, V.P. Business Development] is a high-technology business consultant with a focus on marketing and strategic planning. He earned a B.Sc. (Physics) in the UK and after moving to California in 1968 worked for Silicon Valley companies, including Fairchild Semiconductor, Advanced Micro Devices (AMD), and Altera Corporation, in roles from product marketing engineer to CEO.

My Genesis Article

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Genesis: A Silicon Valley Tale

FAIRCHILD SEMICONDUCTOR CORPORATION

TECH HISTORY ARTICLE

BY DR JEFF DROBMAN

Highlights

- Fairchild founding
- Intel founding
- AMD history
- AMD – Intel rivalry
- Search for CMOS
- RISC CPU Architecture
- Legendary Parties & Conferences
- Anecdotes
- Valley Significant Others
- Genesis org-chart
- Process Technology Evolution
- Anniversaries of Technologies

My Genesis Article

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

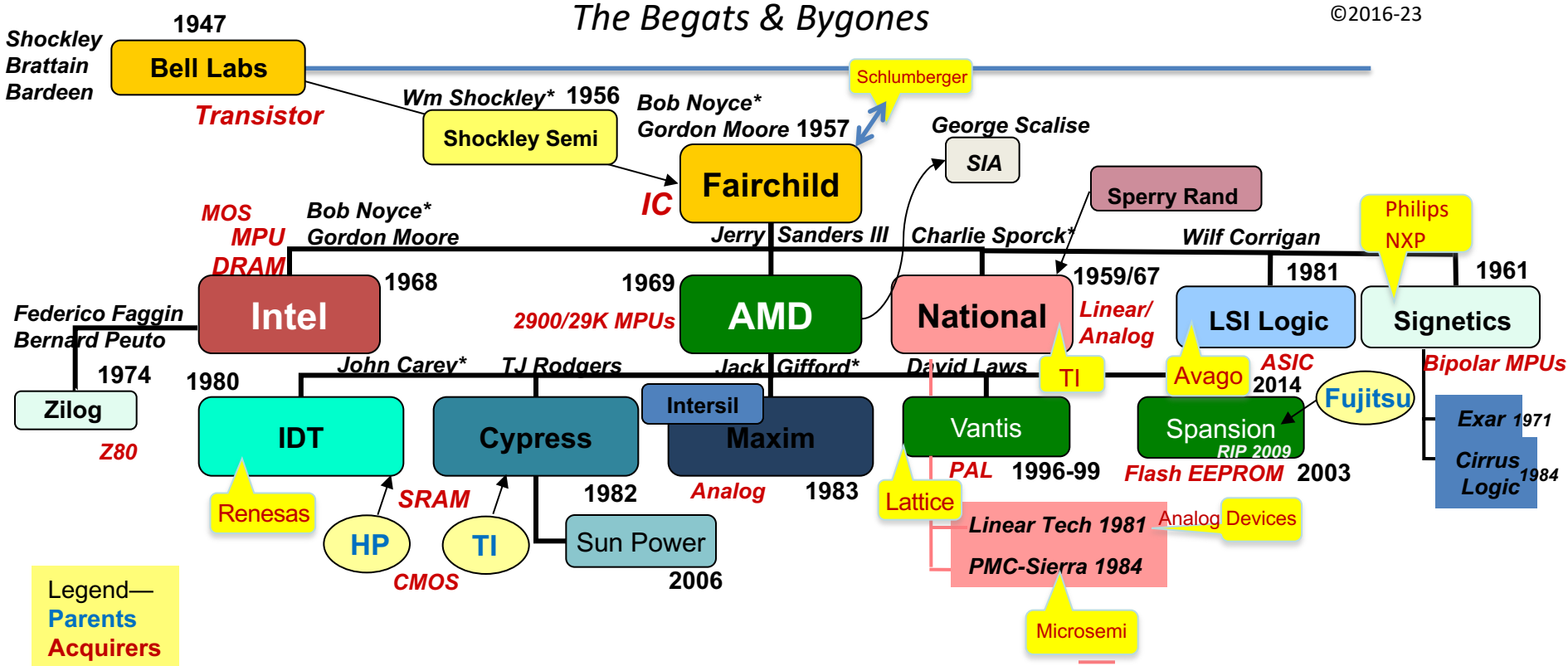
The Legend

It has long been *legendary* that companies in Silicon Valley got started in garages and beach houses, and I am setting the record straight: *It is true*. **Apple** was started in Steve Wozniak's garage, when friend Steve Jobs came by and saw his hobby computer. **Advanced Micro Devices** (AMD) got its start in founding president Jerry Sanders' rented Malibu beach house, on a chilly December evening in 1968 – though the house was heated considerably by those entrepreneurial fires. AMD was incorporated 5 months later (May 1969).

Genesis of Silicon Valley

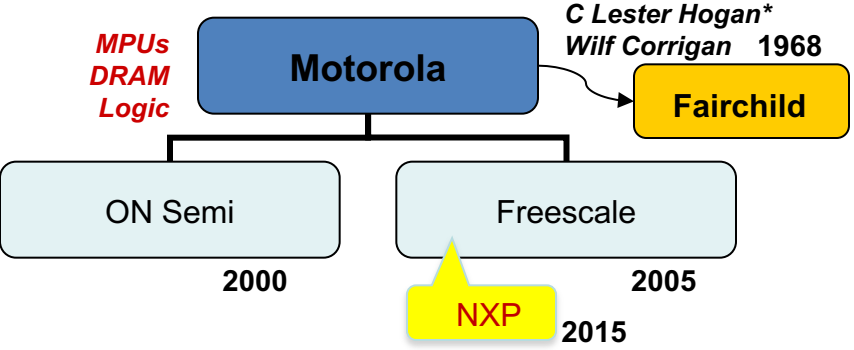
The Begats & Bygones

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

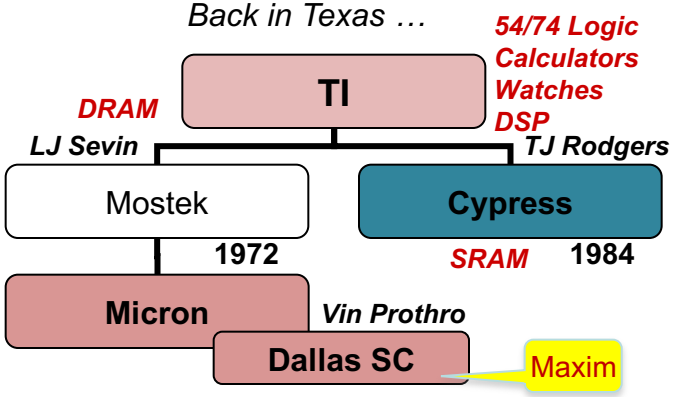


Meanwhile…

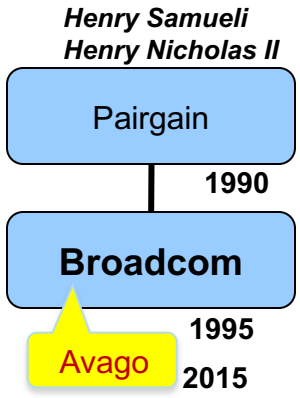
Back in Arizona …



Back in Texas …



Much later in SoCal …



DSJ Dr Jeff DR JEFF SOFTWARE INDIE APP DEVELOPER
Jeff Drobman
©2016-23

https://www.eetimes.com/the-new-silicon-frontier-chapter-4-startup-fever-and-venture-capital/

DESIGNLINES | EE LIFE

The New Silicon Frontier Chapter 4: Startup Fever and Venture Capital

MELTING POT FOR THE FAIRCHILDREN

Sheldon Roberts, Eugene Kleiner, and Jean Hoerni's collective decision to leave and compete against Fairchild, just over three years after the company was founded, was the first of what would be many subsequent defections and spinouts, eventually known as "Fairchildren," directly or indirectly creating dozens of corporations, including Intel and AMD. In doing so, Fairchild sowed the seeds of innovation across multiple companies in the region that would eventually become known as Silicon Valley.

While it is unclear who came up with the moniker, "Silicon Valley," Don Hoefler, a technology reporter for the industry publication *Electronic News*, is often credited with popularizing the name in a 1971 column about the region's chip industry. Hoefler also promoted the area's innovative qualities, and was one of the first writers to chronicle the Northern Californian technology industry as a community.

Don Hoefler

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Local watering holes, restaurants and other hot spots provided venues for Silicon Valley's "work hard, play hard" ethos, where industry folk gathered after work to drink, gossip, brag, trade war stories, talk shop, exchange ideas, change jobs and develop new contacts. Key venues included the Wagon Wheel, Lion & Compass, and Ricky's, along with the Peppermill and the Sunnyvale Hilton.

DSJ Dr Jeff DR JEFF SOFTWARE INDIE APP DEVELOPER
Jeff Drobman
©2016-23

THE FAIRCHILD LEGACY

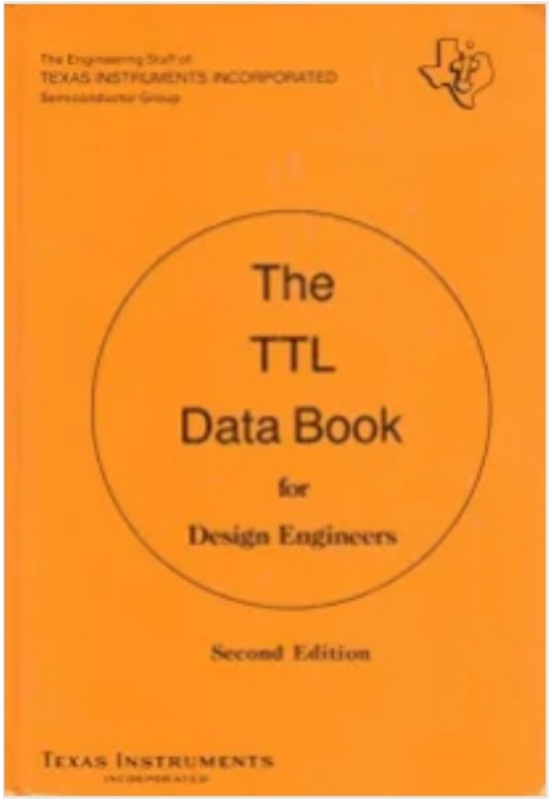
Throughout the first half of the 1960s, Fairchild was the undisputed semiconductor leader, forging ahead across all industry segments, be it design, technology, production or sales. Early sales and marketing efforts were modest and military-oriented; that changed in 1961 when Robert Noyce and Tom Bay recruited a group of aggressive salesmen and marketing specialists, including Jerry Sanders III and Floyd Kvamme. The newcomers transformed Fairchild's sales and marketing departments into one of the industry's legends.

Among the pivotal moments was Fairchild's entry into the consumer TV market. Attracted by potential high volumes, Sanders wanted to replace the tube (valve) CRT driver with a transistor, but the target price was U.S. $1.50. Transistors at that time were selling to the military for $150.00. In what can only be regarded as a massive leap of faith, Noyce's instructions to Sanders were, "Go take the order, Jerry. We'll figure out how to do it later. Maybe we'll have to build it in Hong Kong and put it in plastic, but right now let's just do it."

EE Times 50 1972 2022

TI TTL

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



The TTL Data Book for Design Engineers.

By always ensuring any bill of materials included at least one TTL part that was only available from it, Texas Instruments was able to stay one step ahead of the competition and own the T'TL market for the best part of 30 years, until standard logic eventually fell victim to the 1980s application-specific IC revolution.

DSJ Dr Jeff DR JEFF SOFTWARE INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Charles Sporck, Noyce's operations manager often credited with running the industry's tightest ship, left in early 1968 along with Pierre Lamond to join Widlar and Talbert at National Semiconductor. That triggered Noyce and Moore's departure from the firm later that same year–a pivotal moment in the eventual demise of the firm. The collective exodus of Sporck, Noyce, and Moore, along with so many other executives, signaled the end of an era, prompting Sherman Fairchild to bring in a new management team, led by C. Lester Hogan, then vice president of Motorola Semiconductor.

Sporck → National

HOGAN'S HEROES

Hogan's arrival, and the subsequent displacement of Fairchild managers, demoralized the firm even further, prompting a further exodus of employees who would launch a host of new companies. Leading a group dubbed "Hogan's Heroes," the ultra-conservative Motorola executives immediately clashed with Sanders, Fairchild's flamboyant sales chief.

Hogan/Wilf/Sanders

DSJ Dr Jeff DR JEFF SOFTWARE INDIE APP DEVELOPER
Jeff Drobman
©2016-23

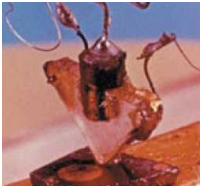
While initially slow to respond to the changing market under Sander's direction, Fairchild embarked on a strategy of leapfrogging Texas Instruments by focusing on more complex large scale, 30-plus gate parts, instead of simpler small and medium scale devices under 30 gates — a strategy that was proving popular and successful with engineers. The move forced Texas Instruments to recognize the threat and copy all of Fairchild's 9300 series parts under 74 series numbers (for example the 9300 became the 74195 and the 9341 the 74181.)

Sander's entire strategy collapsed, however, when Hogan capitulated to Ken Olsen, founder and CEO of Digital Equipment Corporation and a key Fairchild customer. Olsen wanted Fairchild to give up on its proprietary TTL technology and instead second-source Texas Instruments' 74 Series TTL. Against Sanders' wishes, Hogan agreed, signing the death warrant for Fairchild's TTL strategy. Sanders was, understandably livid. "You've just killed the company, Ken," Sander's fumed.

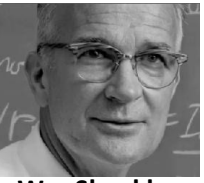
Hogan's betrayal was the last straw for Sanders. He, together with a group of Fairchild engineers, quit to start Advanced Micro Devices. With Sanders installed as president, one of his first moves was to establish the mantra: "People first, revenues and profits will follow." Sanders also gave every employee stock options in the new company, an innovation at the time.

Founders HoF

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



Bell Labs

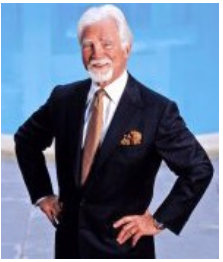


Wm Shockley

Fairchild founders (8)

Fairchild Chairman/CEO, LSI Logic founder

Wilf Corrigan



Jerry Sanders

CEO, AMD
1969–2002

Jack Gifford

AMD co-founder

In 1983, Gifford co-founded Maxim Integrated Products

Bob Noyce

Gordon Moore

Cypress Semi founder

Intel

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

1968

Intel Originals

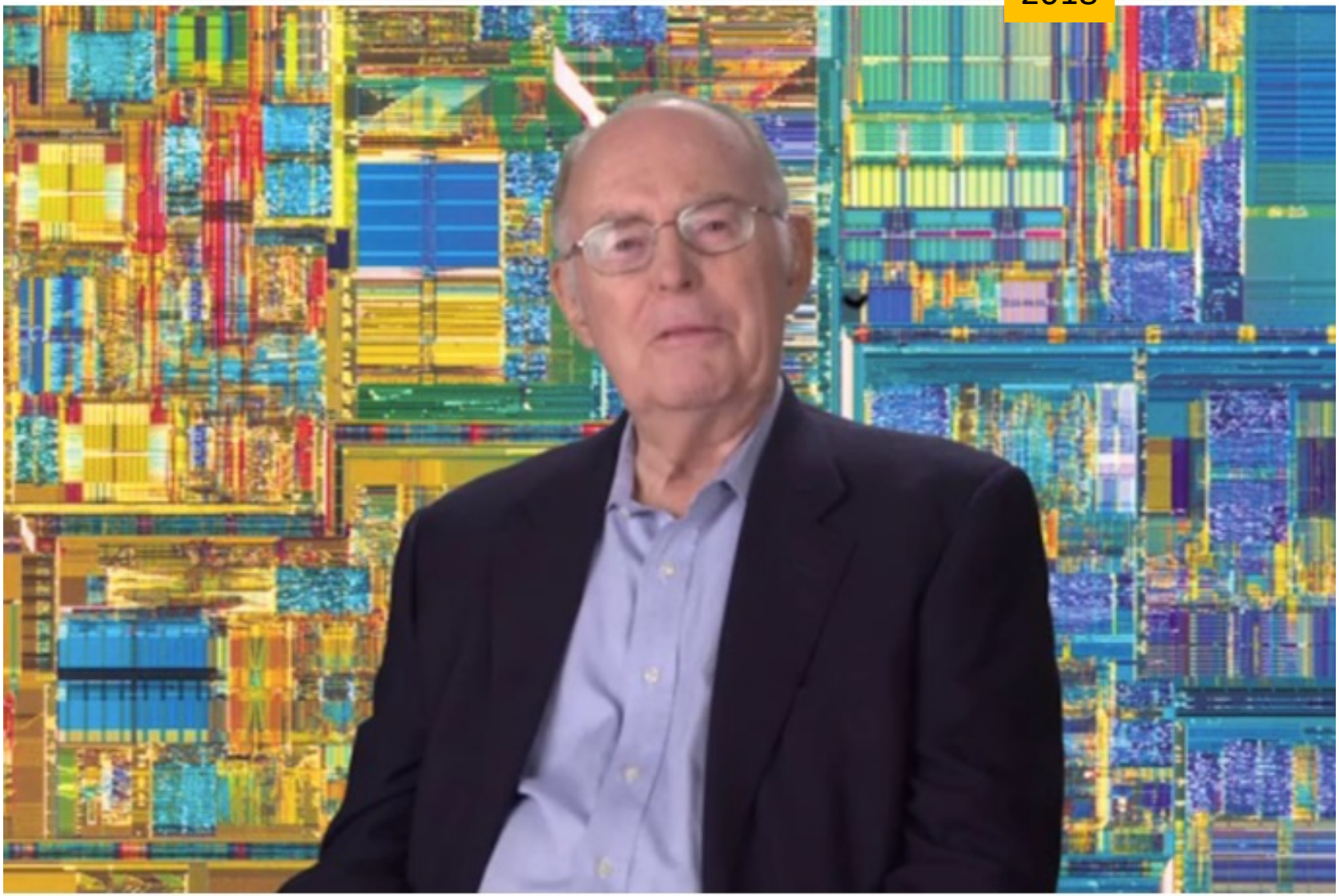
L to R:
Andy Grove
Bob Noyce
Gordon Moore

Founders:
Bob Noyce
Gordon Moore

Intel CEO Gordon Moore

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

2018



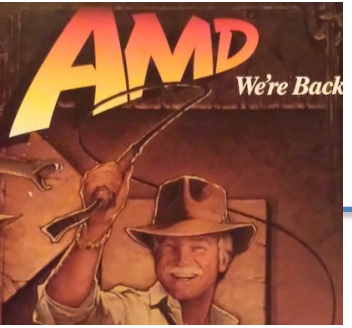
Screenshot of Gordon Moore featured in Scientists You Must Know *by the Science History Institute.* Courtesy of the Science History Institute.

Intel CEO Andy Grove

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

AMD

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



1969

From left: W. Jerry Sanders III, President and Chairman of the Board. D. John Carey, Managing Director of Complex Digital Operations. Sven E. Simonsen, Director of Engineering, Complex Digital Operations. Frank T. Botte, Director of Development, Analog Operations. James N. Giles, Director of Engineering, Analog Operations. Edwin J. Turney, Director of Sales and Administration. Jack F. Gifford, Director of Marketing and Business Development. R. Lawrence Stenger, Managing Director, Analog Operations.

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Motorola was a major chip company, having pioneered in digital logic and microprocessors such as the 6800/68000 and PPC. but Motorola no longer exists. it was first split into 2 companies: Motorola Solutions and Motorola Mobility, in 2011 (sold to Google in 2012, then to Lenovo in 2014). that was after the chip business was split up: ON Semi in 1999, and then Freescale in 2004, which was then sold to NXP Semi (Philips) in 2015.

Fate	Demerged into Motorola Mobility and Motorola Solutions in 2011
Successors	Motorola Mobility Motorola Solutions NXP Semiconductors ON Semiconductor CommScope (General Instrument) Cambium Networks
Founded	September 25, 1928; 92 years ago
Founders	Paul and Joseph Galvin
Defunct	January 4, 2011; 10 years ago

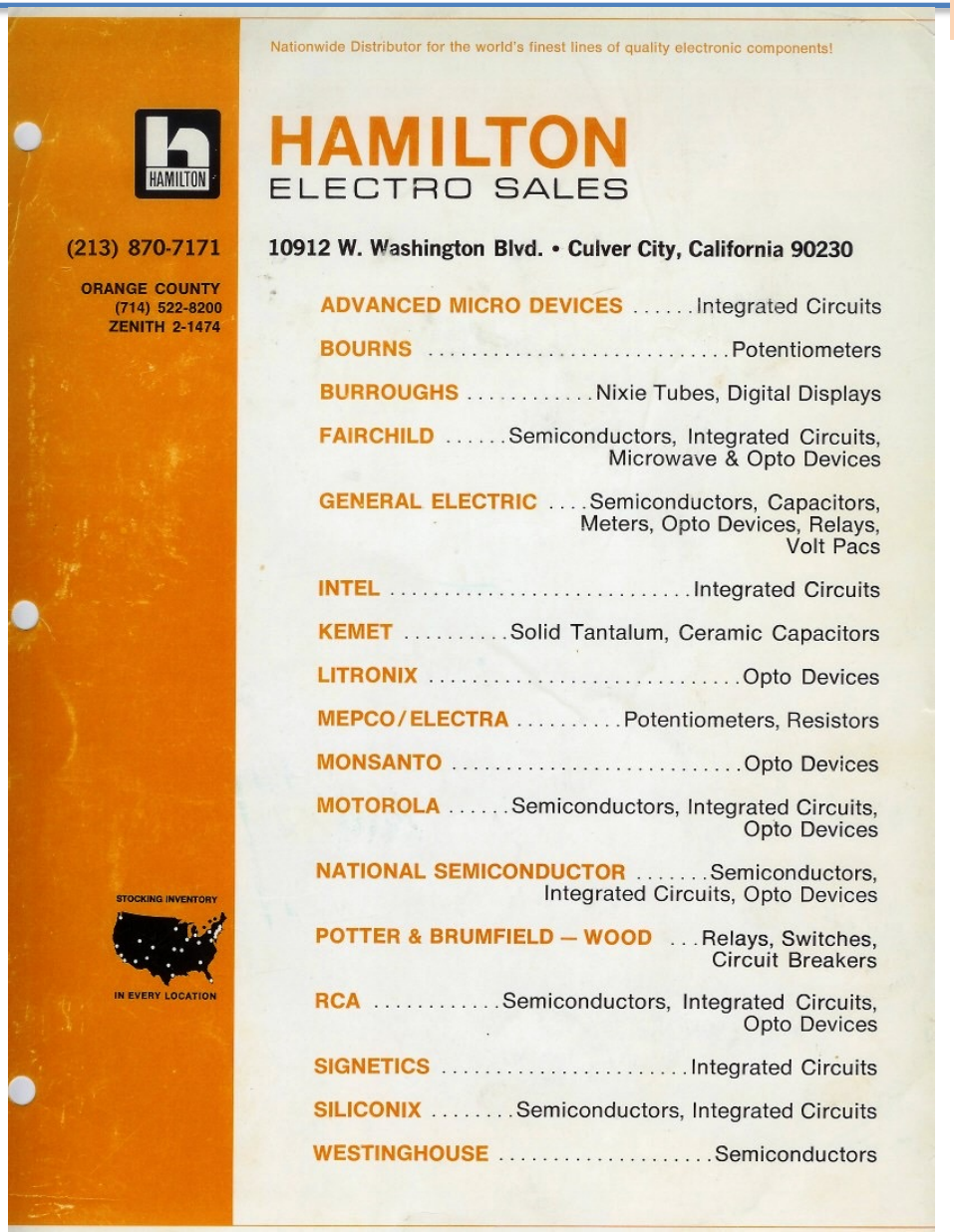
from Wikipedia:

Motorola, Inc. (/ˌmoʊtəˈroʊlə/[2]) was an American multinational telecommunications company founded on September 25, 1928, based in Schaumburg, Illinois. After having lost $4.3 billion from 2007 to 2009, the company demerged into two independent public companies, Motorola Mobility and Motorola Solutions on January 4, 2011.[3] Motorola Inc. was renamed Motorola Solutions and is legally the direct successor to the original company after the demerger from Motorola Mobility.[4][5] Motorola Mobility was sold to Google in 2012, and acquired by Chinese company Lenovo in 2014

Electronic Device Cos.

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

1972



Nationwide Distributor for the world's finest lines of quality electronic components!

HAMILTON
ELECTRO SALES

(213) 870-7171

ORANGE COUNTY
(714) 522-8200
ZENITH 2-1474

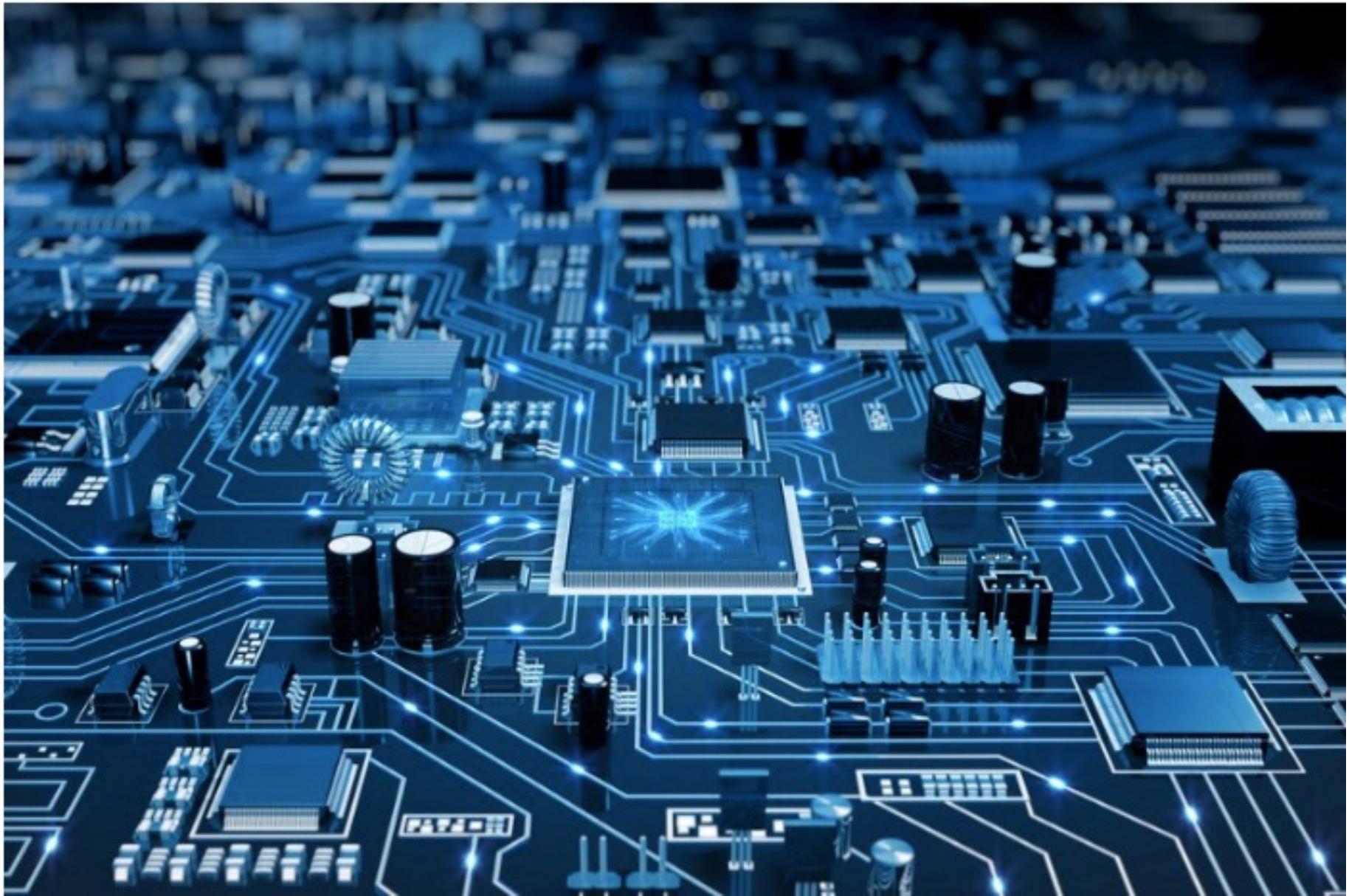
10912 W. Washington Blvd. • Culver City, California 90230

- ADVANCED MICRO DEVICES ..... Integrated Circuits
- BOURNS ..... Potentiometers
- BURROUGHS ..... Nixie Tubes, Digital Displays
- FAIRCHILD ..... Semiconductors, Integrated Circuits, Microwave & Opto Devices
- GENERAL ELECTRIC .... Semiconductors, Capacitors, Meters, Opto Devices, Relays, Volt Pacs
- INTEL ..... Integrated Circuits
- KEMET ..... Solid Tantalum, Ceramic Capacitors
- LITRONIX ..... Opto Devices
- MEPCO/ELECTRA ..... Potentiometers, Resistors
- MONSANTO ..... Opto Devices
- MOTOROLA ..... Semiconductors, Integrated Circuits, Opto Devices
- NATIONAL SEMICONDUCTOR ..... Semiconductors, Integrated Circuits, Opto Devices
- POTTER & BRUMFIELD — WOOD ... Relays, Switches, Circuit Breakers
- RCA ..... Semiconductors, Integrated Circuits, Opto Devices
- SIGNETICS ..... Integrated Circuits
- SILICONIX ..... Semiconductors, Integrated Circuits
- WESTINGHOUSE ..... Semiconductors

STOCKING INVENTORY
IN EVERY LOCATION

Chips on a PCB

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



Dev Boards

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

POPULAR DEVELOPMENT BOARDS

SoCKit Development Kit

The SoCKit Development Kit presents a robust hardware design platform built around the Altera System-on-Chip (SoC) FPGA, which combines the latest dual-core Cortex-A9 embedded cores with industry-leading programmable logic for ultimate design flexibility

MAX1000 IOT Maker Board

For engineers designing compact smart solutions for the IoT market, this FPGA IoT Maker Board is an excellent tool to speed up the development process and enter the market with a high-performance, reliable product.

Google Coral Dev Board

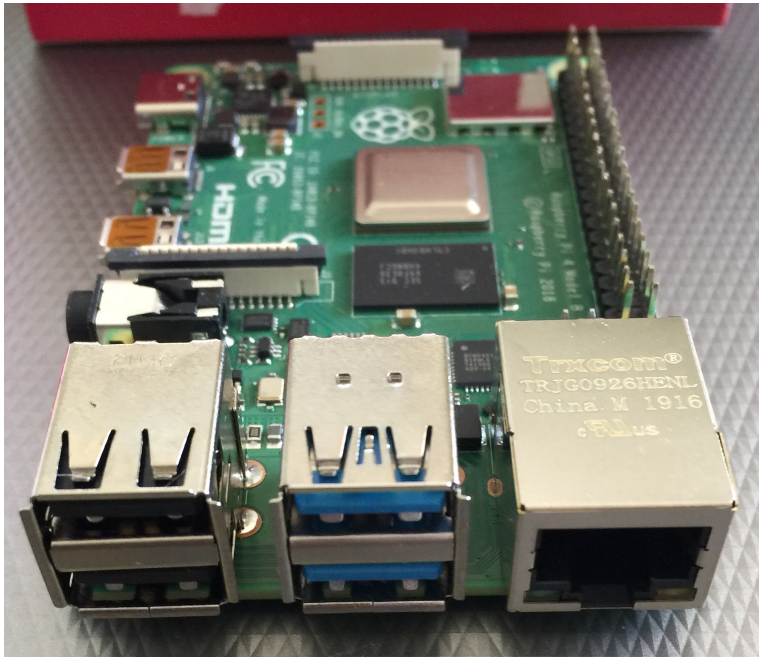
The Coral Dev Board is now in-stock and available for free 1-day shipping at Arrow.com. Prototype, scale, and deploy with more flexibility using the Coral Dev Board and accessories with Google.

ARM-A72 Development Board

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Raspberry Pi 4

Your tiny, dual-display, desktop computer



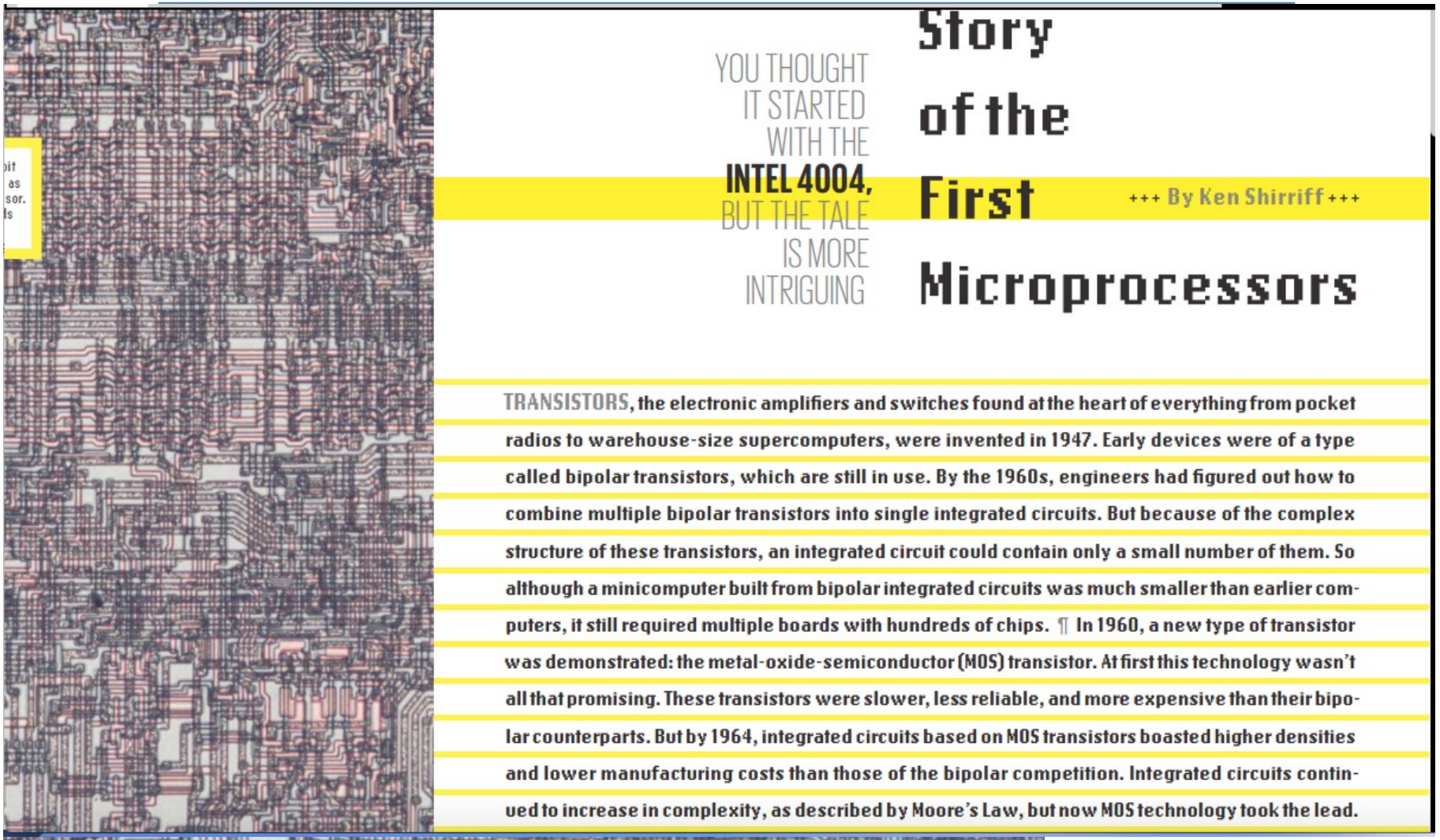
Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Microprocessors
MPU/MCU

Early MPUs

Jeff Drobman
©2016-23



YOU THOUGHT IT STARTED WITH THE **INTEL 4004,** BUT THE TALE IS MORE INTRIGUING

Story of the First Microprocessors

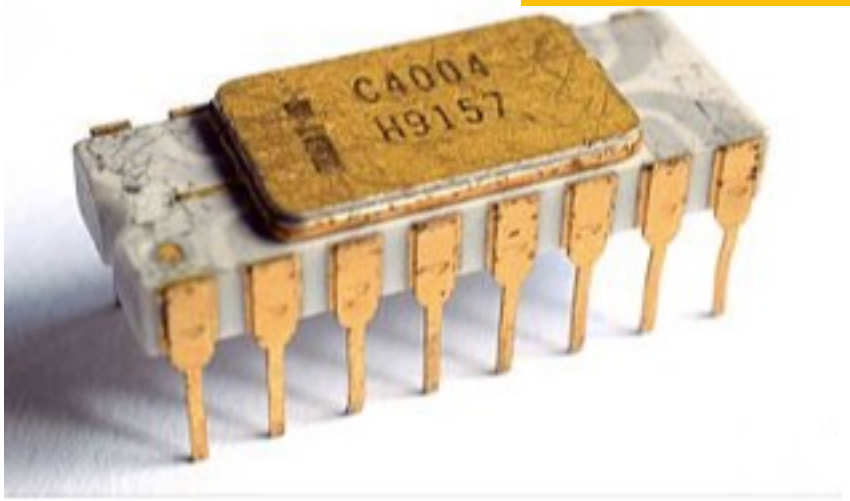
+++ By Ken Shirriff +++

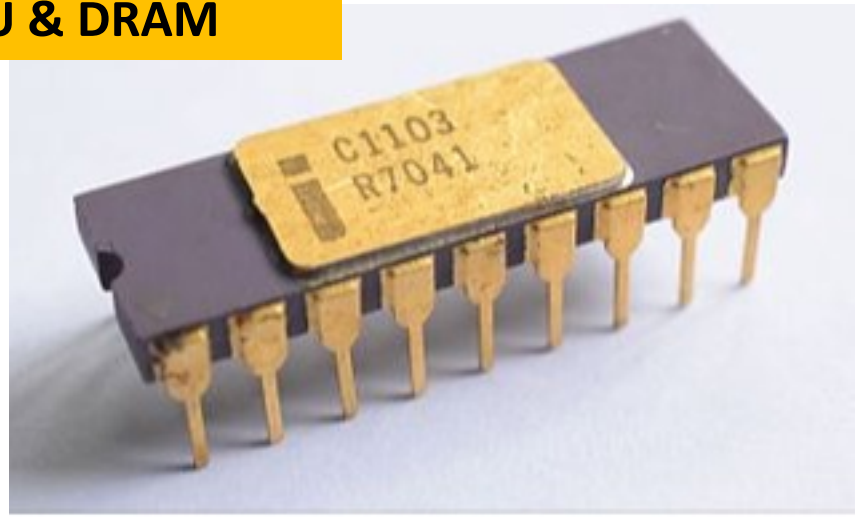
TRANSISTORS, the electronic amplifiers and switches found at the heart of everything from pocket radios to warehouse-size supercomputers, were invented in 1947. Early devices were of a type called bipolar transistors, which are still in use. By the 1960s, engineers had figured out how to combine multiple bipolar transistors into single integrated circuits. But because of the complex structure of these transistors, an integrated circuit could contain only a small number of them. So although a minicomputer built from bipolar integrated circuits was much smaller than earlier computers, it still required multiple boards with hundreds of chips. ¶ In 1960, a new type of transistor was demonstrated: the metal-oxide-semiconductor (MOS) transistor. At first this technology wasn't all that promising. These transistors were slower, less reliable, and more expensive than their bipolar counterparts. But by 1964, integrated circuits based on MOS transistors boasted higher densities and lower manufacturing costs than those of the bipolar competition. Integrated circuits continued to increase in complexity, as described by Moore's Law, but now MOS technology took the lead.

Intel i4004 and i1103A

Jeff Drobman
©2016-23

World's 1st MPU & DRAM





The **Intel 4004** is a 4-bit central processing unit (CPU) released by Intel Corporation in 1971. It was the first commercially available microprocessor, and the first in a long line of Intel CPUs. The chip design, implemented with the MOS silicon gate technology, started in April 1970, and was created by Federi

1st **MPU**	**4-bit**
1971	

The **1103** is a dynamic random-access memory (DRAM) integrated circuit (IC) developed and fabricated by Intel. Introduced in October 1970, the 1103 was the first commercially available DRAM IC; and due to its small physical size and low price relative to magnetic-core memory, it replaced the la

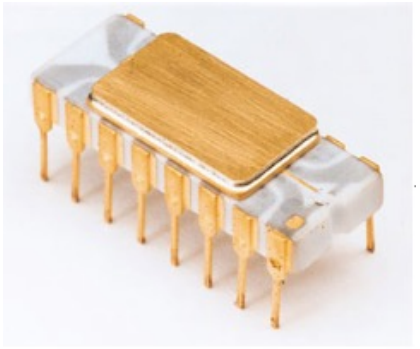
1st **DRAM**	1K x1
1970	

© 2022 Dr Jeff Drobman – drjeff@drjeffsoftware.com

Intel i8008

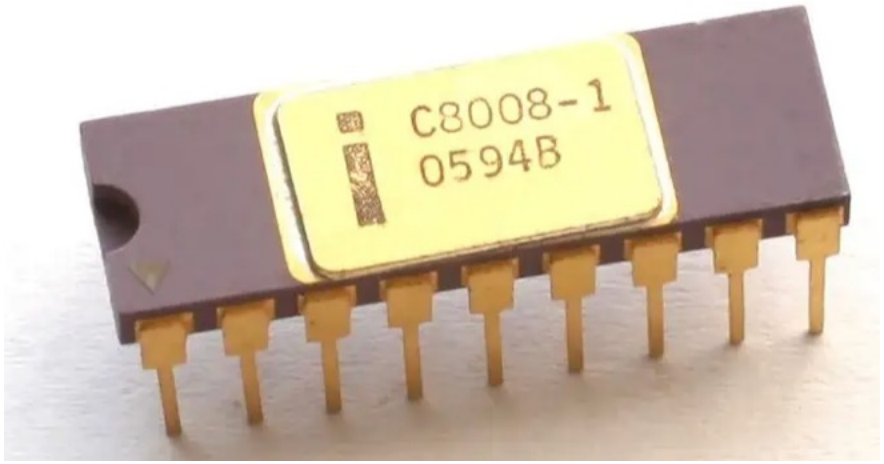
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

World's 1st 8-bit MPU



The venerable 16-pin side-brazed DIP.
(Click image to view full size)

1971 **DIP**



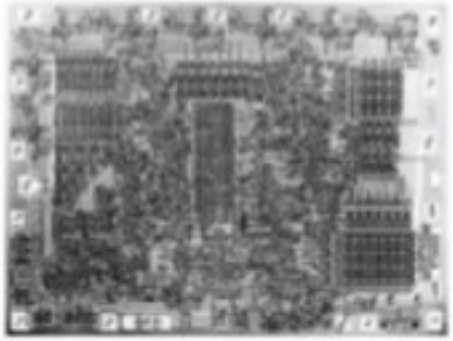
© 2022 Dr Jeff Drobman – drjeff@drjeffsoftware.com

Intel i4004/i8008

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

World's 1st 4/8-bit MPU's

4004
2300 transistors



8008
3098 transistors

Intel MCS-4 and MCS-8 design team and CPU chips

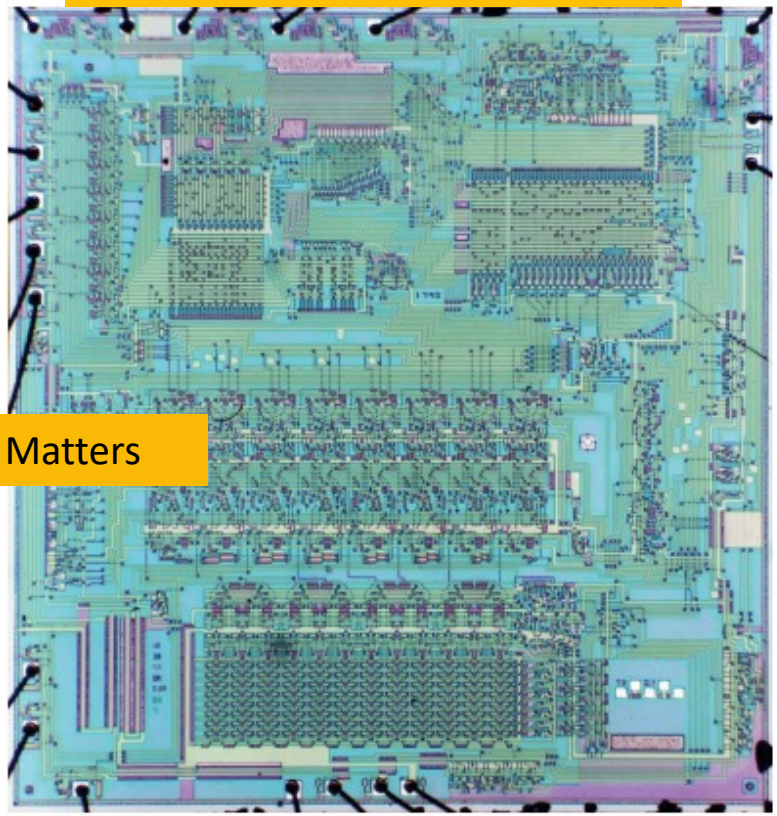
© 2022 Dr Jeff Drobman – drjeff@drjeffsoftware.com

Intel i4004/i8008

Jeff Drobman
©2016-23

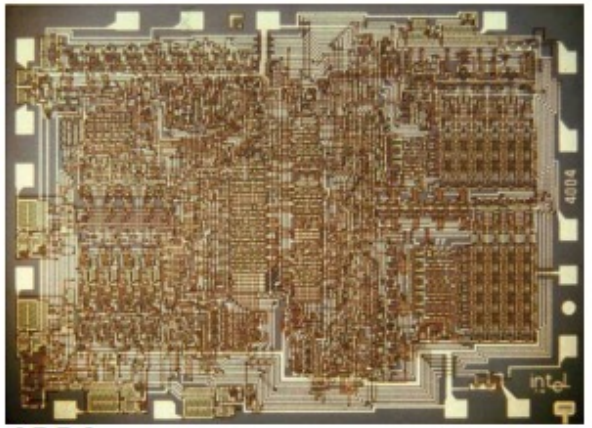
World's 1st 4/8-bit MPU's

Die Size Matters

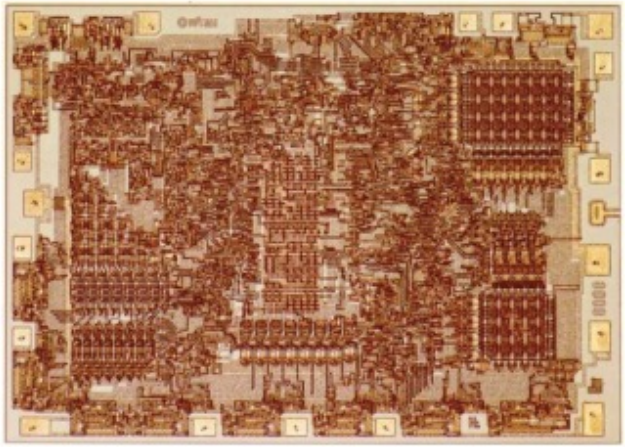


TMX 1795
3,078 transistors

1 mm



4004
2,300 transistors



8008
3,098 transistors

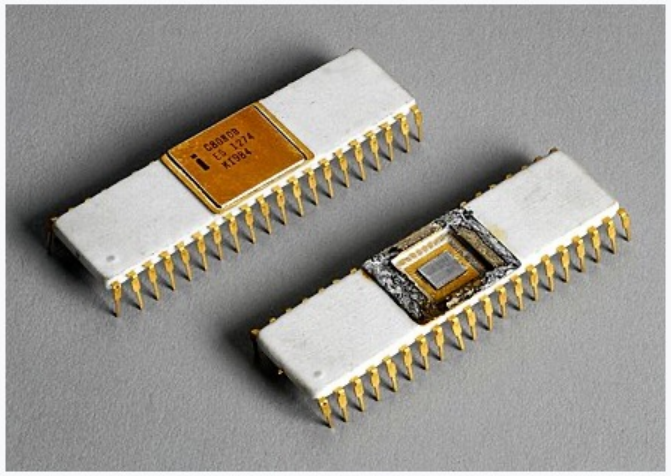
EVERYTHING'S BIGGER IN TEXAS: Although Texas Instruments' TMX 1795 and Intel's 8008 had a similar number of transistors, the former required a much larger silicon die. Indeed, the TMX 1795 was larger than the Intel 8008 and 4004 combined. Intel's engineers believed that its large size made the TI chip impractical to produce in commercial quantities, but TI's very successful TMS 0100 calculator chip, introduced at about the same time, had an even larger die. So the connection between die size and commercial viability must not have been straightforward.

8-bit i8080

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wikipedia

Intel 8080



Closed and open Intel 8080 processor

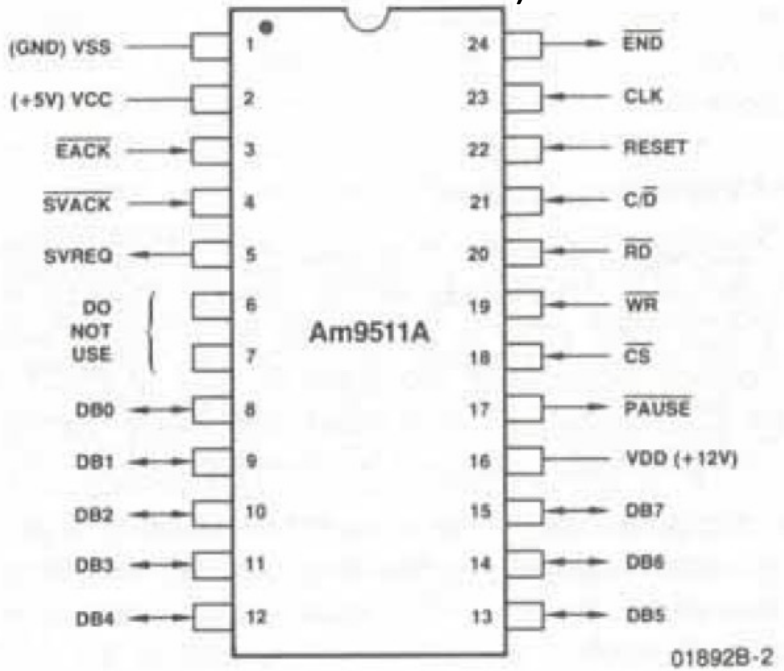
General information

Launched	April 1974; 49 years ago
Discontinued	1990; 33 years ago[1]
Max. CPU clock rate	2 MHz to 3.125 MHz
Technology node	6 μm
Transistors	4,500 or 6,000[

AMD 8080 clone (Am9080)

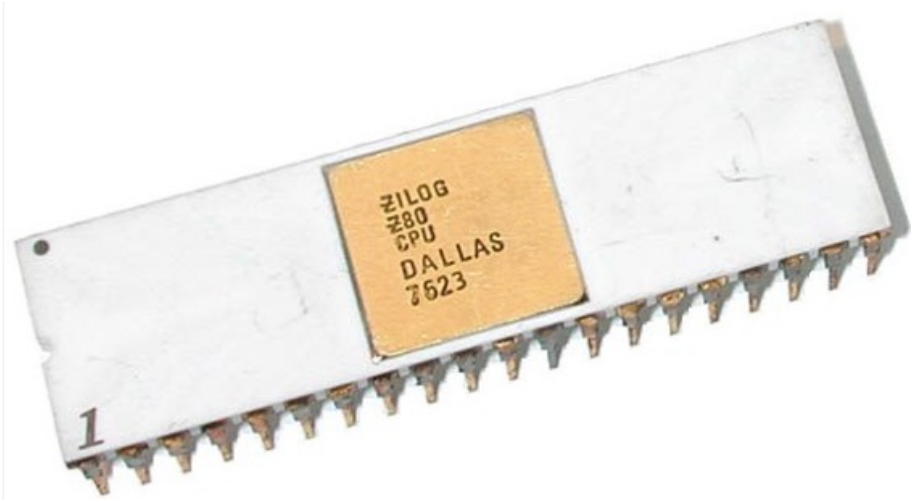
CONNECTION DIAGRAM
Top View
D-24-2

AMD Am9511 FPU)



Venerable Zilog Z80

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



- Improved i8080 (dual register bank, for one)
- Spinoff of Intel (Federico Faggin, Bernard Peuto, et al.

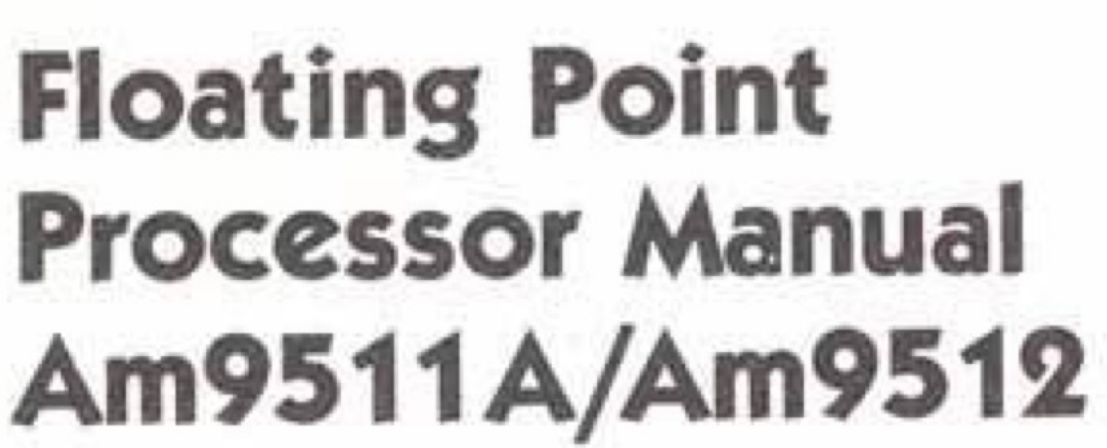
AMD FPU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Am9511/12

From AMD datasheet

Single & Double precision **Floating-poin**t with *transcendentals*



AMD FPU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Am9511/12

From AMD datasheet

Single & Double precision **Floating-poin**t with *transcendentals*

5.2 Am9511A ARITHMETIC PROCESSOR

This pioneer single-chip arithmetic processor interfaces with most popular 8-bit microprocessors such as Am9080A, Am8085, MC6800 by Motorola and Z80 by Zilog. It can also be used for 16-bit microprocessors such as AmZ8000,* but its performance with such 16-bit microprocessors is somewhat hindered by its 8-bit external data bus.

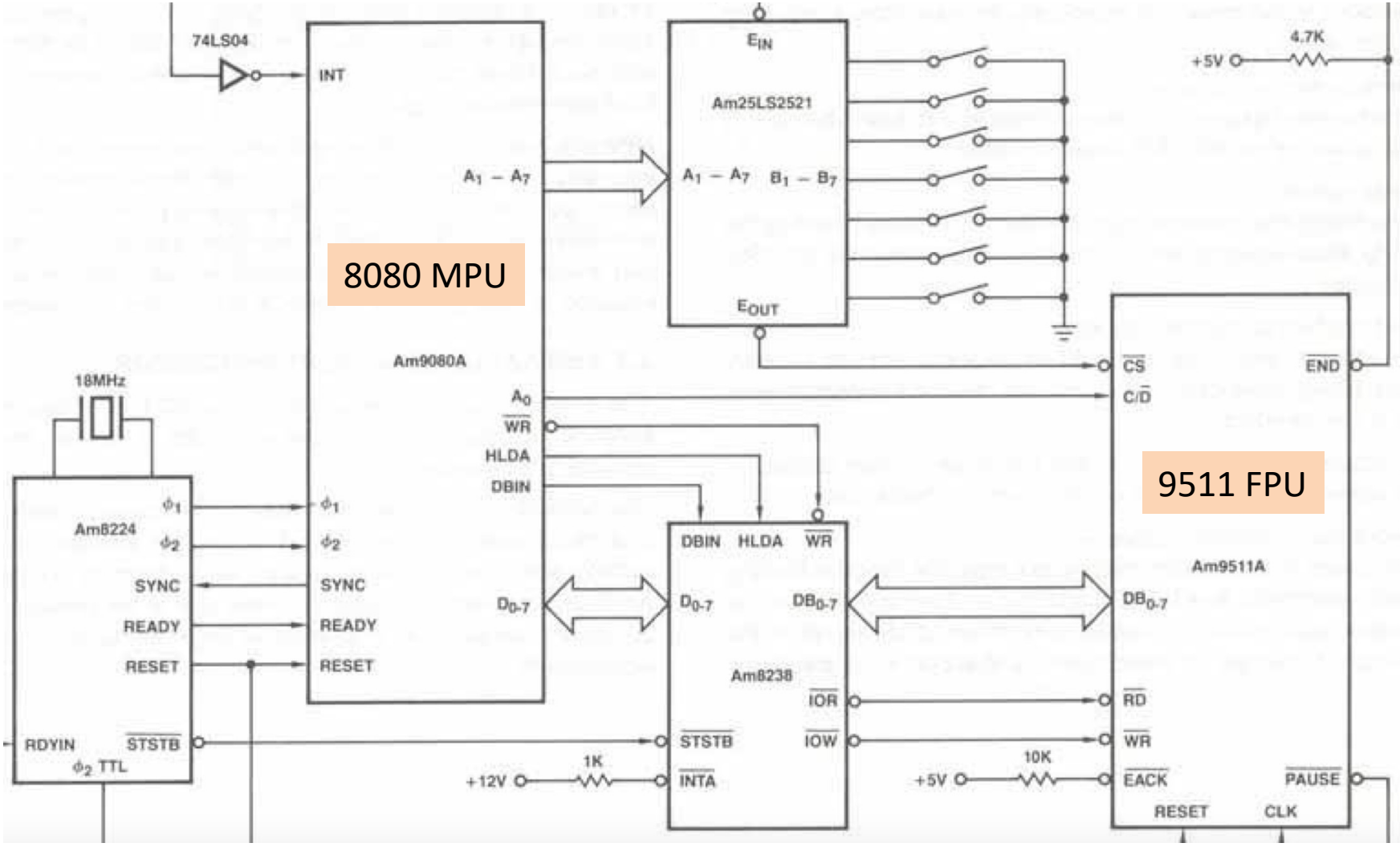
Although the external interface is only 8 bits wide, the Am9511A internally is a 16-bit microprogrammed, stack-oriented floating point machine. It includes not only floating point operations but fixed point as well. In addition to the basic add, subtract, multiply and divide operations, transcendental derived functions are also included. A data sheet of Am9511A is included in Appendix A.

AMD FPU Interface

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Am9511/12

From AMD datasheet

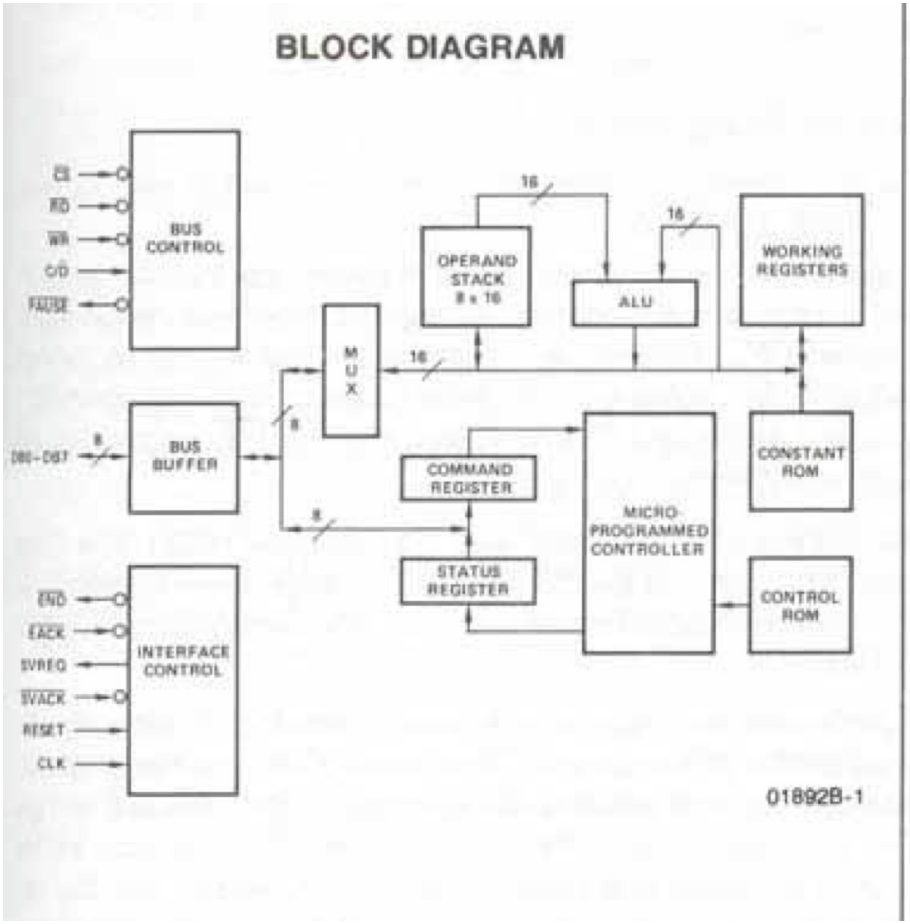


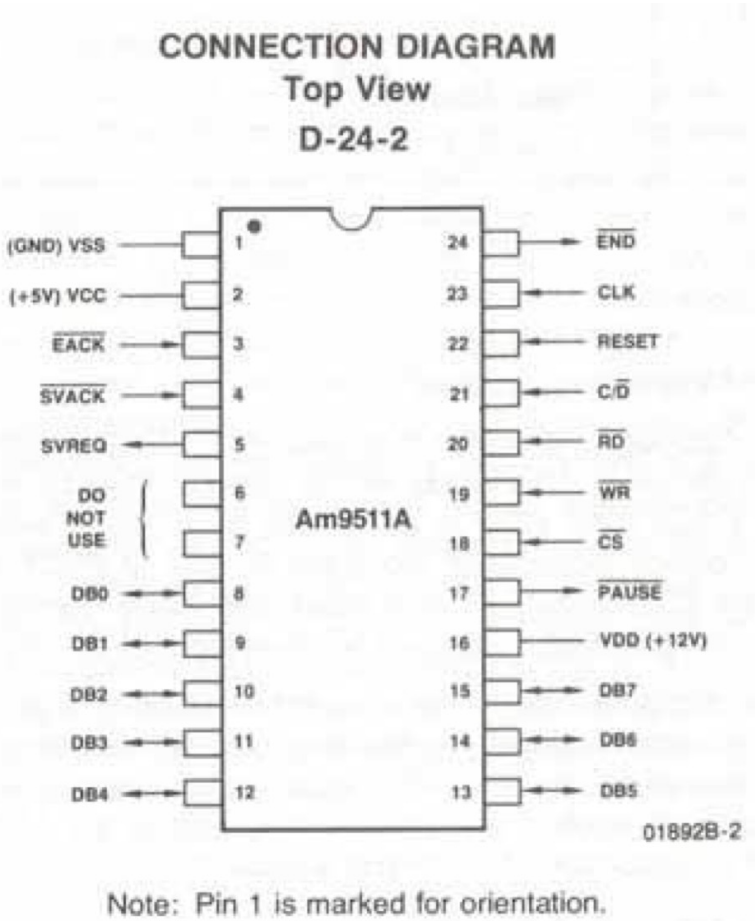
AMD FPU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Am9511/12

From AMD datasheet





Note: Pin 1 is marked for orientation.

AMD FPU Format

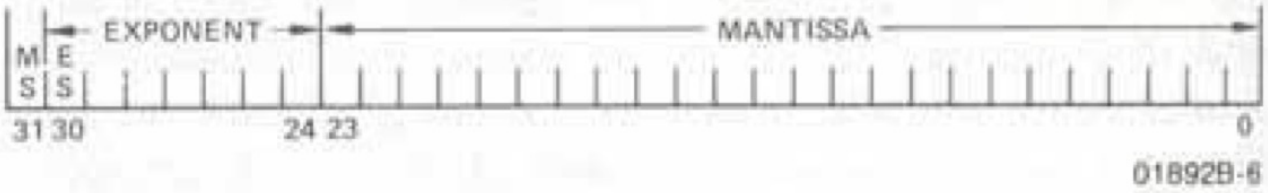
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Am9511/12

From AMD datasheet

FLOATING POINT FORMAT

The format for floating-point values in the Am9511A is given below. The mantissa is expressed as a 24-bit (fractional) value; the exponent is expressed as an unbiased two's complement 7-bit value having a range of −64 to +63. The most significant bit is the sign of the mantissa (0 = positive, 1 = negative), for a total of 32 bits. The binary point is assumed to be to the left of the most significant mantissa bit (bit 23). All floating-point data values must be normalized. Bit 23 must be equal to 1, except for the value zero, which is represented by all zeros.



The range of values that can be represented in this format is $\pm(2.7 \times 10^{-20}$ to $9.2 \times 10^{18})$ and zero.

AMD FPU Instructions

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Am9511/12

From AMD datasheet

Command Mnemonics in Alphabetical Order.

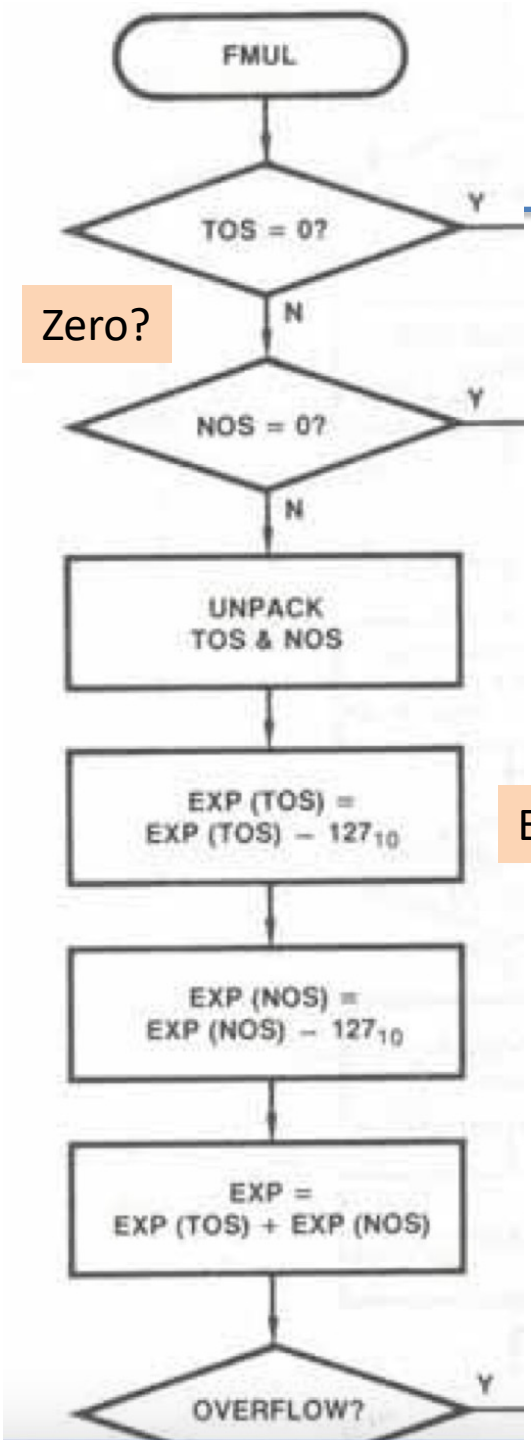
ACOS	ARCCOSINE
ASIN	ARCSINE
ATAN	ARCTANGENT
CHSD	CHANGE SIGN DOUBLE
CHSF	CHANGE SIGN FLOATING
CHSS	CHANGE SIGN SINGLE
COS	COSINE
DADD	DOUBLE ADD
DDIV	DOUBLE DIVIDE
DMUL	DOUBLE MULTIPLY LOWER
DMUU	DOUBLE MULTIPLY UPPER
DSUB	DOUBLE SUBTRACT
EXP	EXPONENTIATION (e^X)
FADD	FLOATING ADD
FDIV	FLOATING DIVIDE
FIXD	FIX DOUBLE
FIXS	FIX SINGLE
FLTD	FLOAT DOUBLE
FLTS	FLOAT SINGLE
FMUL	FLOATING MULTIPLY
FSUB	FLOATING SUBTRACT
LOG	COMMON LOGARITHM
LN	NATURAL LOGARITHM
NOP	NO OPERATION
POPD	POP STACK DOUBLE
POPF	POP STACK FLOATING
POPS	POP STACK SINGLE
PTOD	PUSH STACK DOUBLE
PTOF	PUSH STACK FLOATING
PTOS	PUSH STACK SINGLE
PUPI	PUSH π
PWR	POWER (X^Y)
SADD	SINGLE ADD
SDIV	SINGLE DIVIDE
SIN	SINE
SMUL	SINGLE MULTIPLY LOWER
SMUU	SINGLE MULTIPLY UPPER
SQRT	SQUARE ROOT
SSUB	SINGLE SUBTRACT
TAN	TANGENT
XCHD	EXCHANGE OPERANDS DOUBLE
XCHF	EXCHANGE OPERANDS FLOATING
XCHS	EXCHANGE OPERANDS SINGLE

AMD FPU Flow

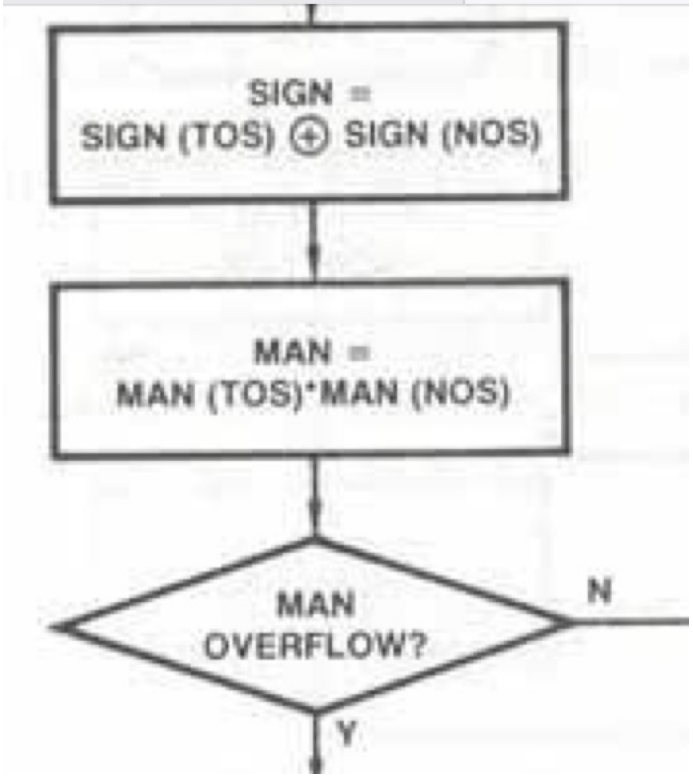
Jeff Drobman
©2016-23

Am9511/12

From AMD datasheet



Mantissa



i8086 History

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

WikiSemi

History of the 8086

The path to the 8086 was not as direct and planned as you might expect. Its earliest ancestor was the Datapoint 2200, a desktop computer/terminal from 1970. The Datapoint 2200 was before the creation of the microprocessor, so it used an 8-bit processor built from a board full of individual TTL integrated circuits. Datapoint asked Intel and Texas Instruments if it would be possible to replace that board of chips with a single chip. Copying the Datapoint 2200's architecture, Texas Instruments created the TMX 1795 processor (1971) and Intel created the 8008 processor (1972). However, Datapoint rejected these processors, a fateful decision. Although Texas Instruments couldn't find a customer for the TMX 1795 processor and abandoned it, Intel decided to sell the 8008 as a product, creating the microprocessor market. Intel followed the 8008 with the improved 8080 (1974) and 8085 (1976) processors. (I've written more about early microprocessors here.)

Datapoint 2200 computer. Photo courtesy of Austin Roche.

i8086 History

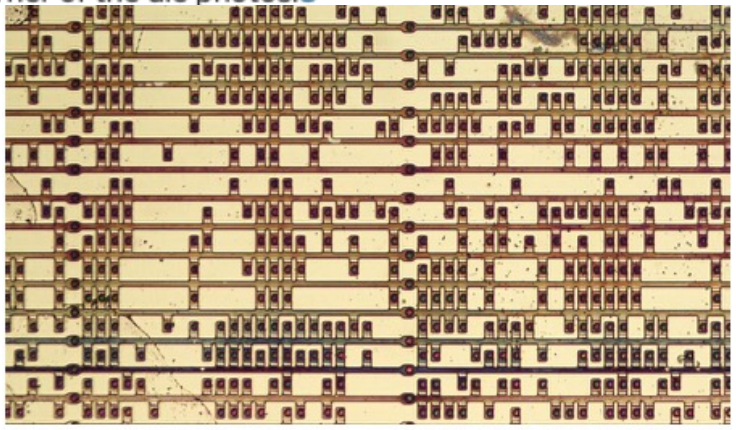
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

WikiSemi

Microcode

One of the hardest parts of computer design is creating the control logic that tells each part of the processor what to do to carry out each instruction. In 1951, Maurice Wilkes came up with the idea of microcode: instead of building the control logic from complex logic gate circuitry, the control logic could be replaced with special code called microcode. To execute an instruction, the computer internally executes several simpler micro-instructions, which are specified by the microcode. With microcode, building the processor's control logic becomes a programming task instead of a logic design task.

Microcode was common in mainframe computers of the 1960s, but early microprocessors such as the 6502 and Z-80 didn't use microcode because early chips didn't have room to store microcode. However, later chips such as the 8086 and 68000, used microcode, taking advantage of increasing chip densities. This allowed the 8086 to implement complex instructions (such as multiplication and string copying) without making the circuitry more complex. The downside was the microcode took a large fraction of the 8086's die; the microcode is visible in the lower-right corner of the die photos.3



A section of the microcode ROM.

i8086 History

DR JEFF SOFTWARE
INDIE APP DEVELOPER
©2016-18 Jeff Drobman
©2016-23

WikiSemi

Why did the IBM PC pick the Intel 8088 processor?7 According to Dr. David Bradley, one of the original IBM PC engineers, a key factor was the team's familiarity with Intel's development systems and processors. (They had used the Intel 8085 in the earlier IBM Datamaster desktop computer.) Another engineer, Lewis Eggebrecht, said the Motorola 68000 was a worthy competitor6 but its 16-bit data bus would significantly increase cost (as with the 8086). He also credited Intel's better support chips and development tools.5

In any case, the decision to use the 8088 processor cemented the success of the x86 family. The IBM PC AT (1984) upgraded to the compatible but more powerful 80286 processor. In 1985, the x86 line moved to 32 bits with the 80386, and then 64 bits in 2003 with AMD's Opteron architecture. The x86 architecture is still being extended with features such as AVX-512 vector operations (2016). But even though all these changes, the x86 architecture retains compatibility with the original 8086.

i8086 16-bit MPU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

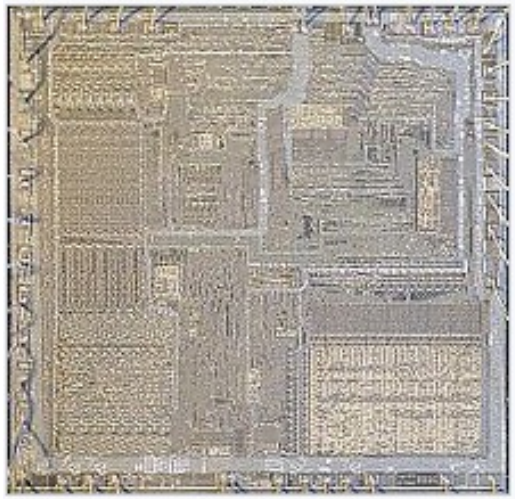
1st 16-bit MPU

1978

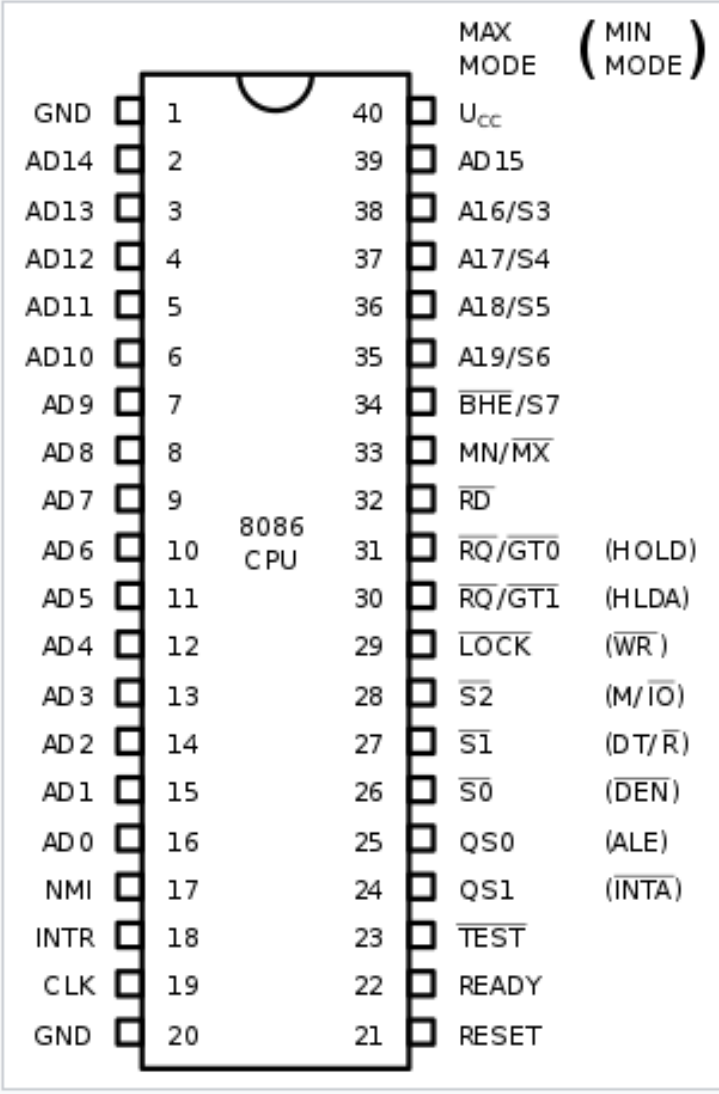
Intel 8086

A rare Intel C8086 processor in purple ceramic DIP package with side-brazed pins

General Info	
Launched	1978
Discontinued	1998[1]
Common manufacturer(s)	Intel, AMD, NEC, Fujitsu, Harris (Intersil), OKI, Siemens AG, Texas Instruments, Mitsubishi, Panasonic (Matsushita)
Performance	
Max. CPU clock rate	5 MHz to 10 MHz
Data width	16 bits
Address width	20 bits
Architecture and classification	
Min. feature size	3 μm
Instruction set	x86-16
Physical specifications	
Transistors	29,000
Co-processor	Intel 8087
Package(s)	40 pin DIP



Intel 8086 CPU die image

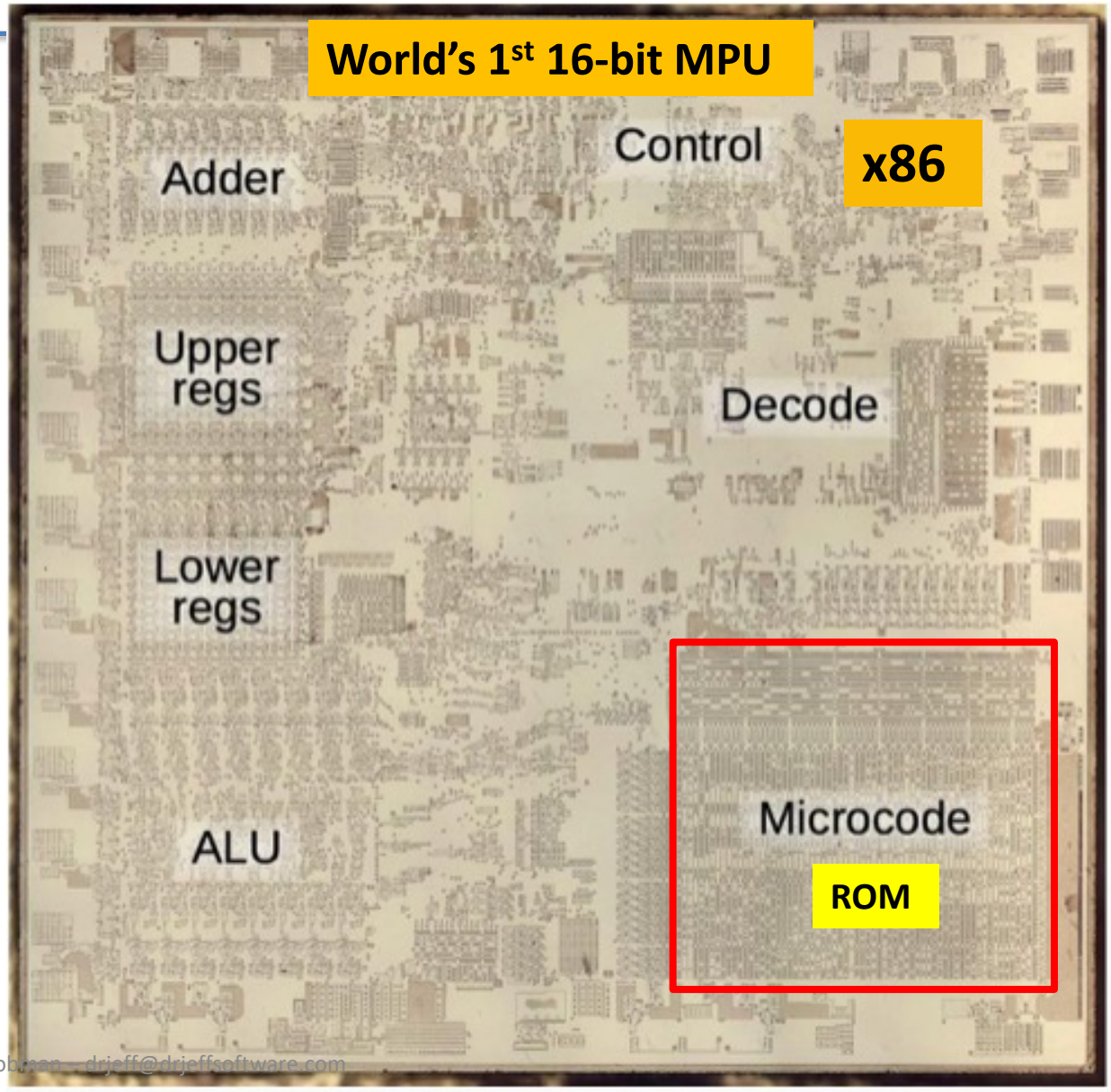


The 8086 pin assignments in min and max mode

Intel i8086 Die

WikiSemi

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

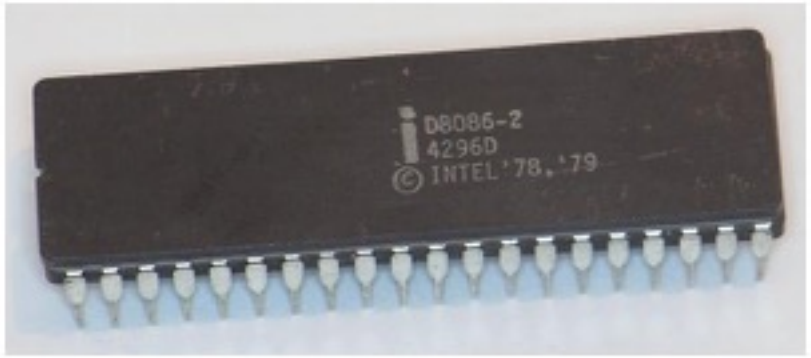


© 2022 Dr Jeff Drobman – drjeff@drjeffsoftware.com

Intel i8086 Package

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

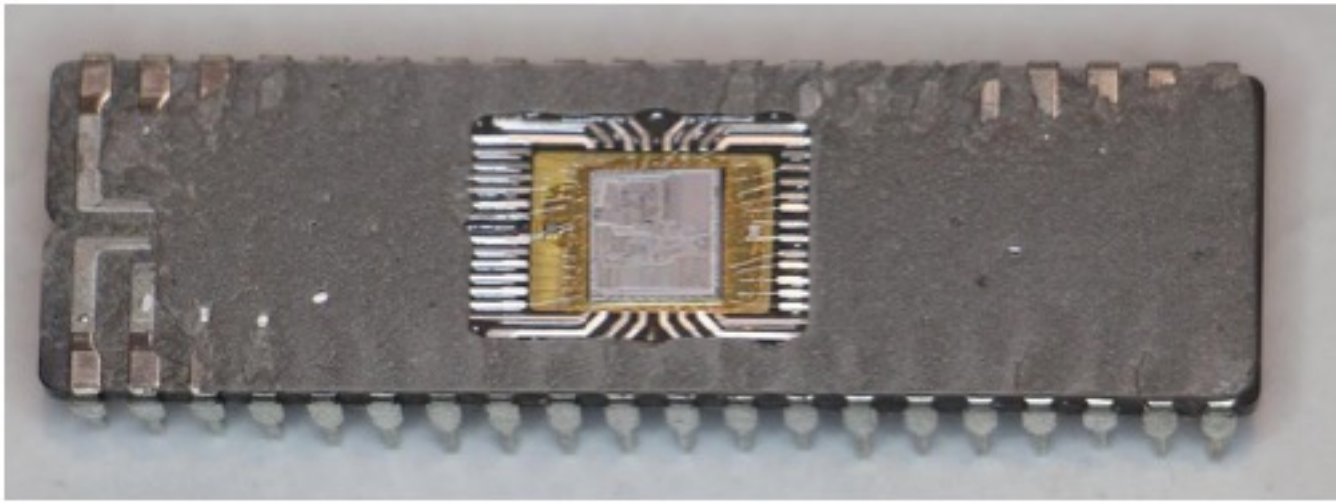
World's 1st 16-bit MPU



WikiSemi

x86

The 8086 chip, in 40-pin ceramic DIP package.

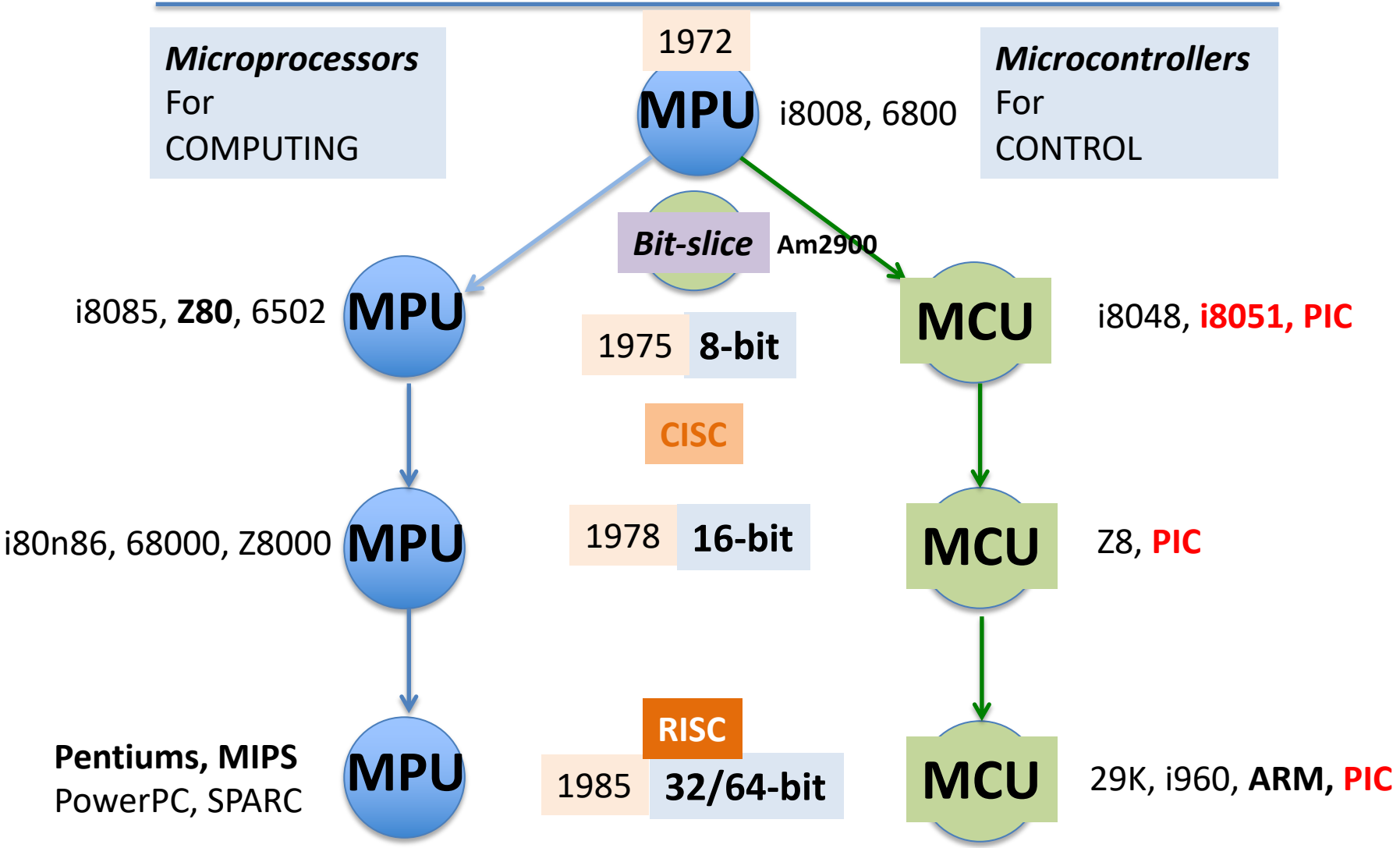


The 8086 die is visible in the middle of the integrated circuit package.

© 2022 Dr Jeff Drobman

MPU/MCU Generations

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



MosTech 6502

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Used in the Apple II

The **MOS Technology 6502** is an 8-bit microprocessor that was designed by a small team led by Chuck Peddle for MOS Technology. The design team had formerly worked at Motorola on the Motorola 6800 project; the 6502 is essentially a simplified, less expensive and faster version of that

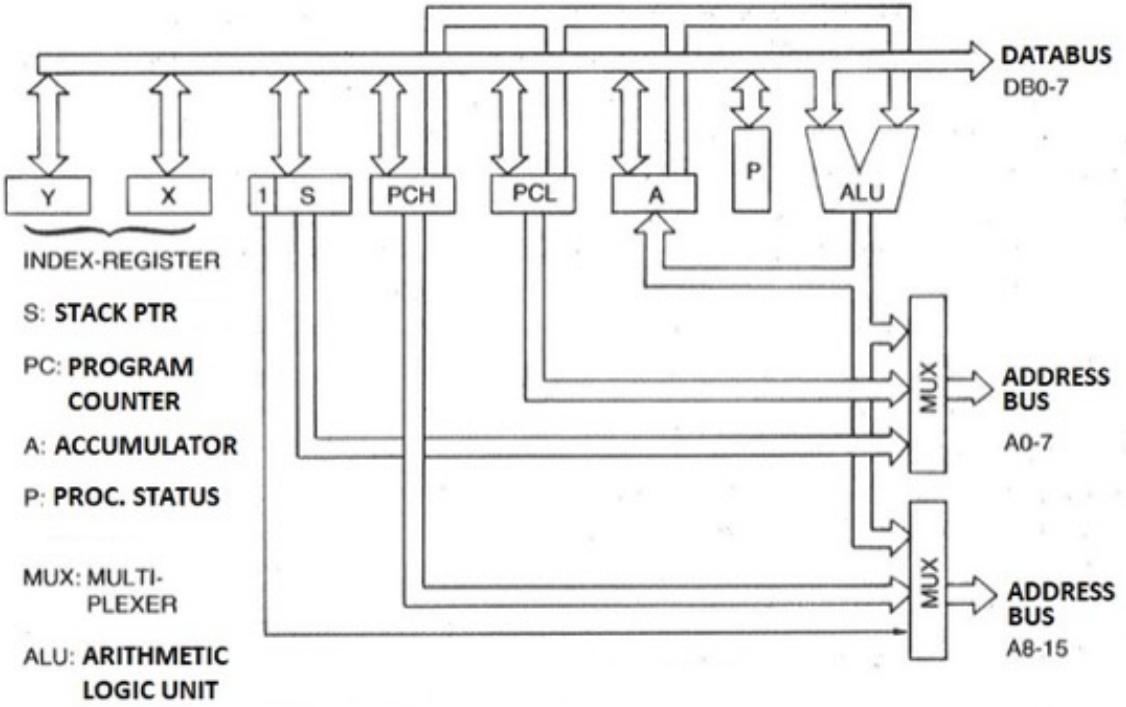
6502 8-bit MPU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Apple II in 1977 | CISC

Other than ALU, what are the basic components of a CPU?

Most of the answers are for more complicated CPUs, with caches, pipelines, DMA etc. But the basic components for a working CPU are much fewer. The 8-bit 6502 microprocessor, introduced in 1975 and used in the Apple][computer and other early personal computers, had only 3510 transistors (compared to the many billions in today's CPUs). Its basic block diagram was fairly simple and easy to understand:



Not shown is the Instruction Register (IR) and decoder logic, which holds the instruction being executed which was fetched from memory.

6502 8-bit MPU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

CISC Apple II in 1977

The 6502 had one 8-bit accumulator, and two 8-bit index registers, 8-bit stack pointer, and a 16-bit program counter so it could address a maximum of 65536 bytes.

MOS 6502 registers

F E D C B A 9 8 7 6 5 4 3 2 1 0 *(bit position, hex)*	
Main registers	
A	**A**ccumulator
Index registers	
X	**X** index
Y	**Y** index
0 0 0 0 0 0 0 1 S	**S**tack **P**ointer
Program counter	
PC	**P**rogram **C**ounter
Status register	
N V - B D I Z C	**P** Processor flags

The high byte of the stack address is hardwired to 1, so stack addresses ranged from 0x1FF (initial value) to 0x100.

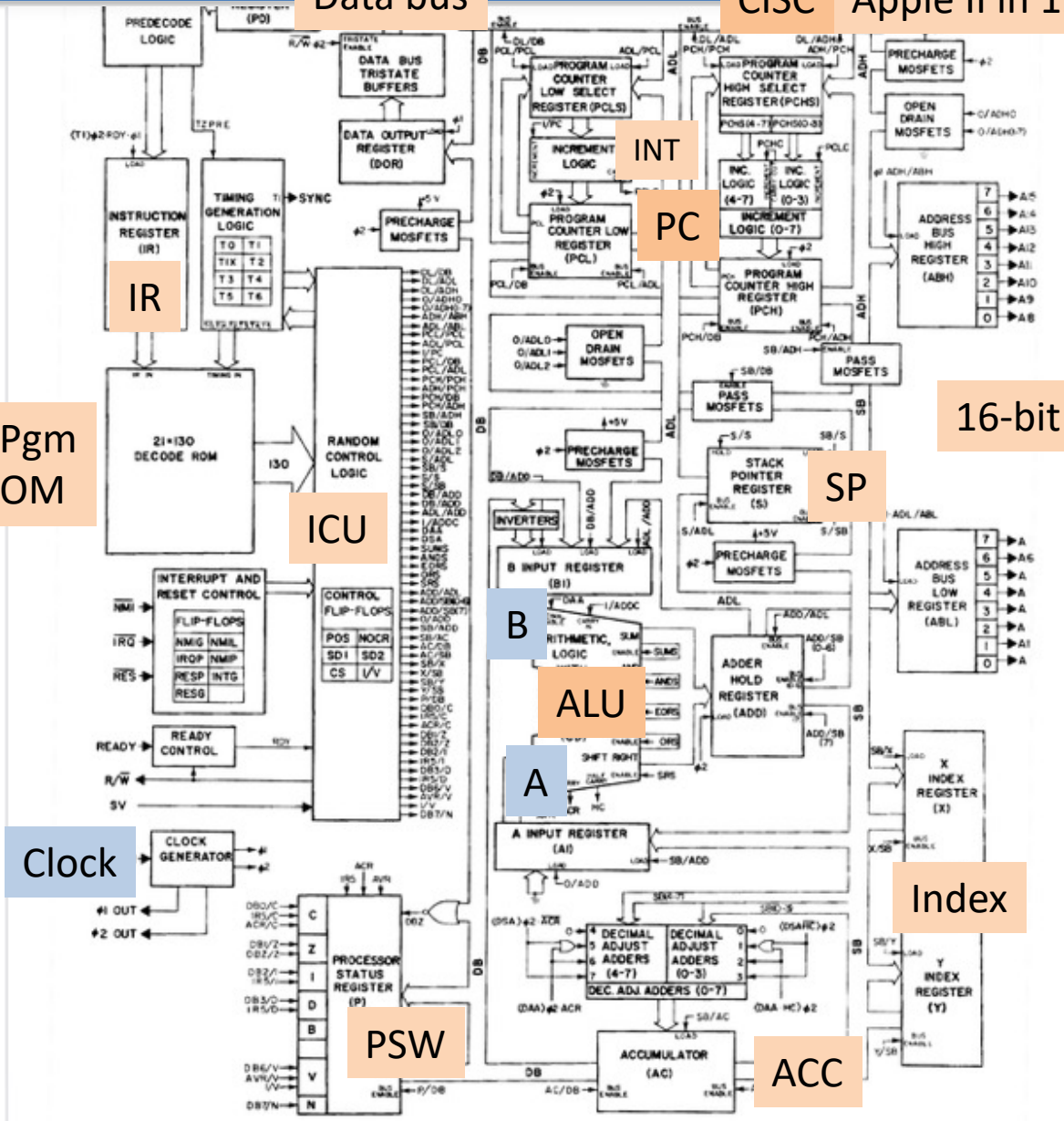
6502 8-bit MPU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Data bus

CISC

Apple II in 1977



INT

PC

IR

uPgm ROM

ICU

16-bit Address bus

SP

B

ALU

A

Clock

Index

PSW

ACC

6502 in 1974

Quora

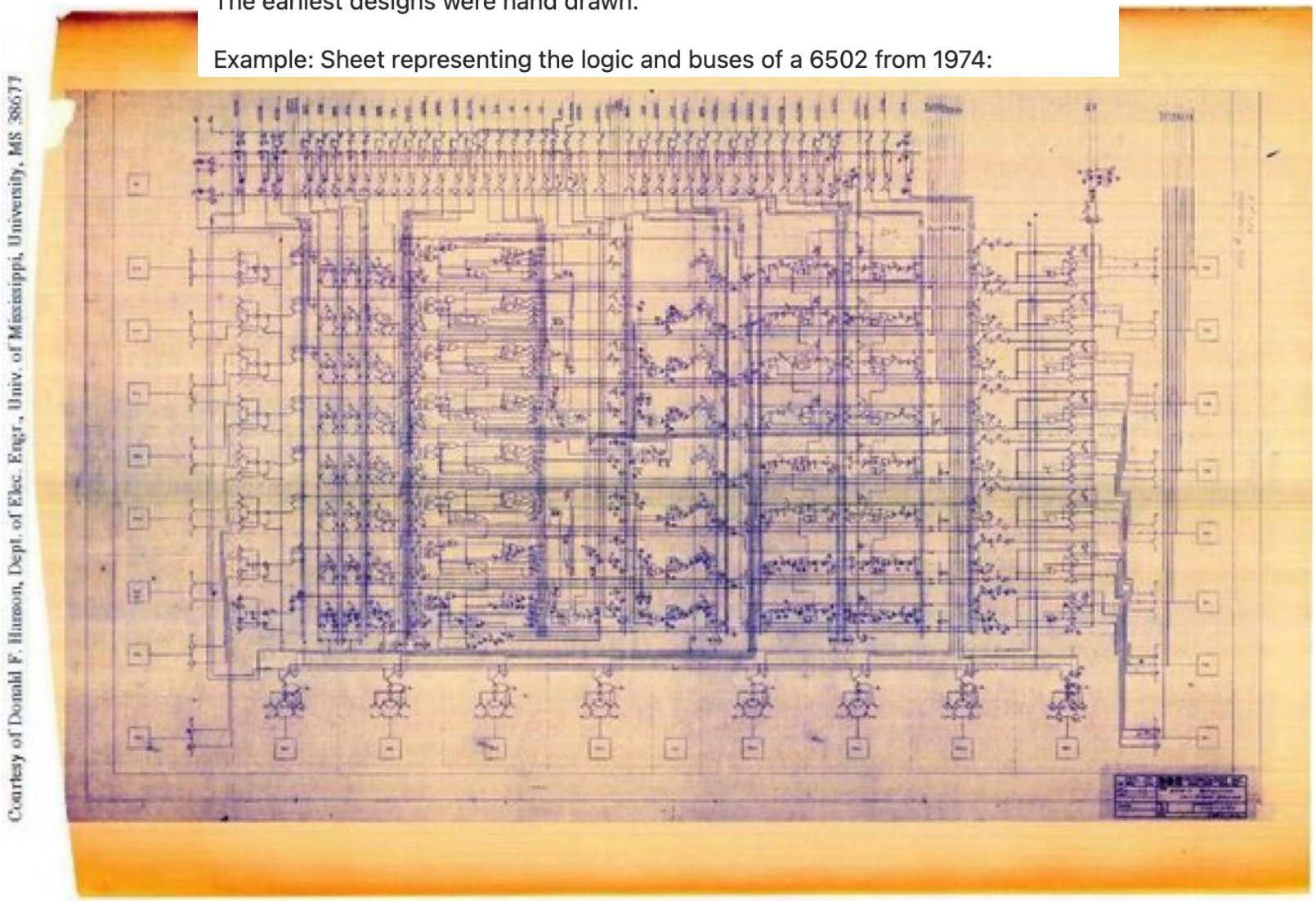
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Yowan Rajcoomar · Follow
IT Engineer (2018–present) · 1y

Related **How were microprocessors designed before Verilog and VHDL?**

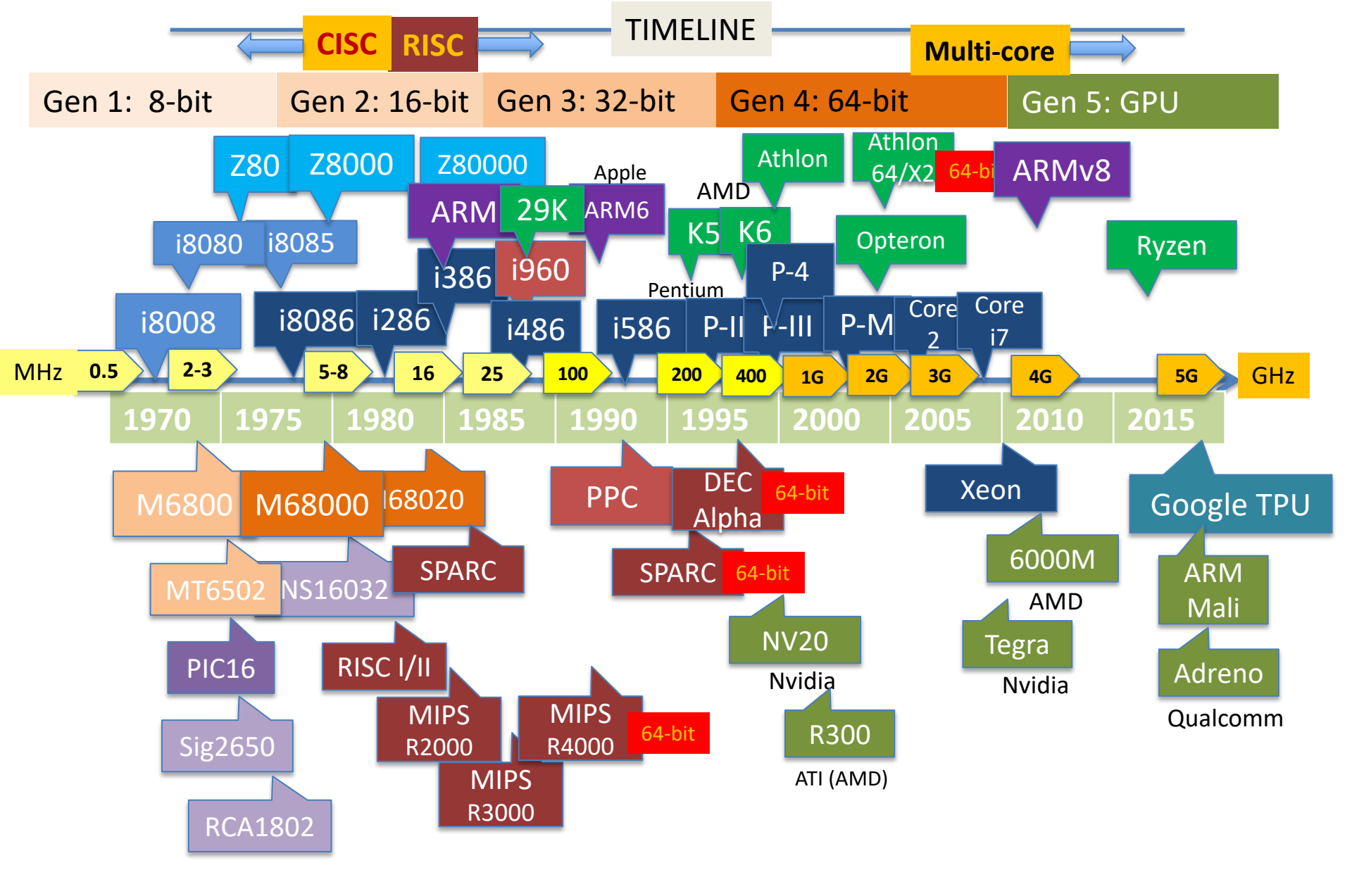
The earliest designs were hand drawn.

Example: Sheet representing the logic and buses of a 6502 from 1974:



MPU Generations

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

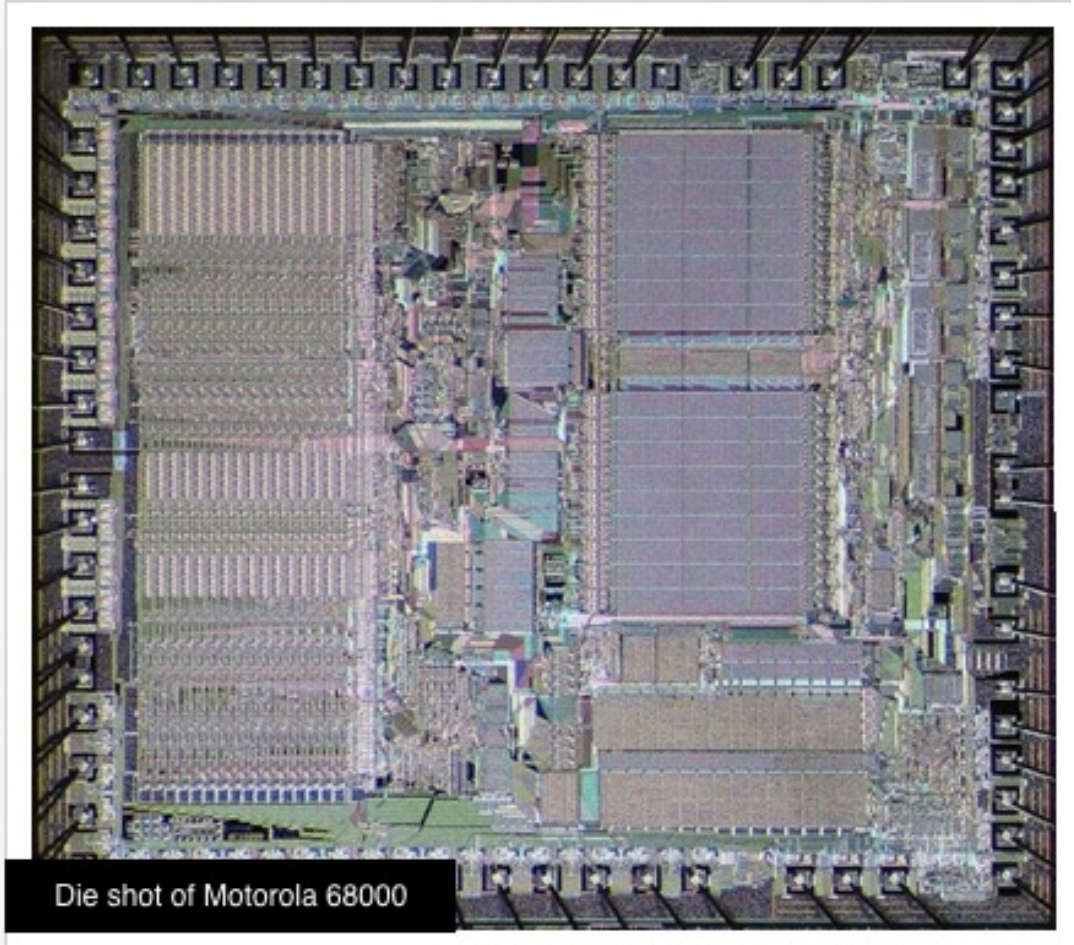


M68000 16-bit MPU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

1980

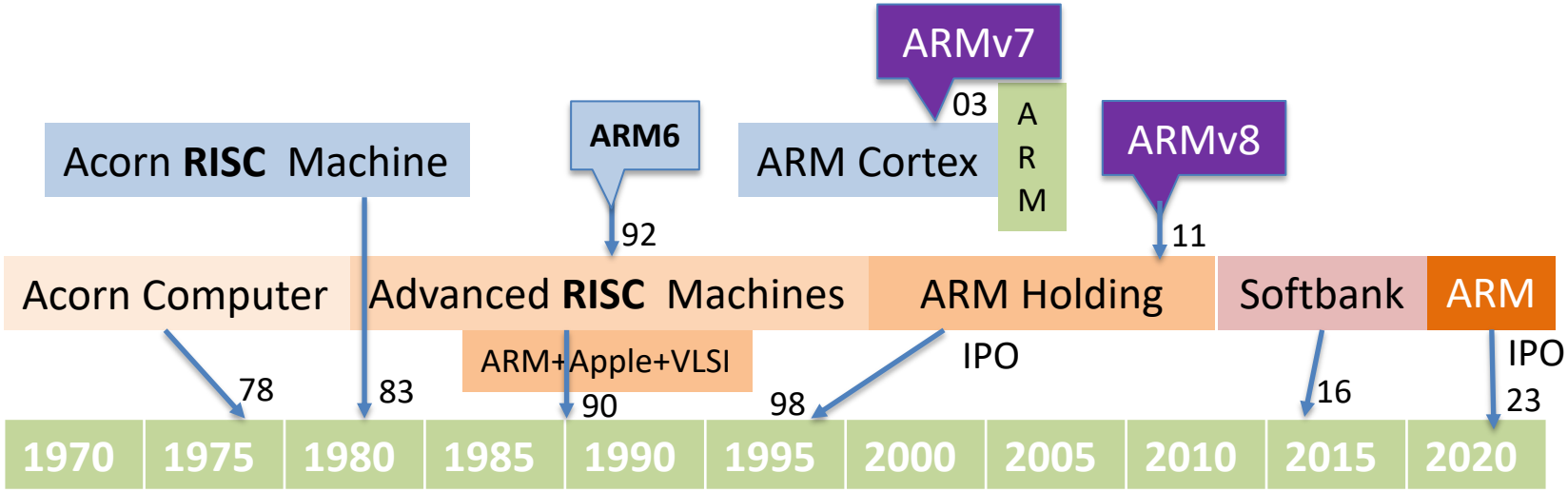
Motorola introduces the 68000 microprocessor



Die shot of Motorola 68000

ARM History

Jeff Drobman
©2016-23



Legend

CPU design

Company

ISA

The **Acorn Archimedes** is a family of personal computers designed by Acorn Computers of Cambridge, England. The systems were based on Acorn's own ARM architecture processors and the proprietary operating systems Arthur and RISC OS. The first models were introduced in 19

ARM Core & ISA Timeline

1985

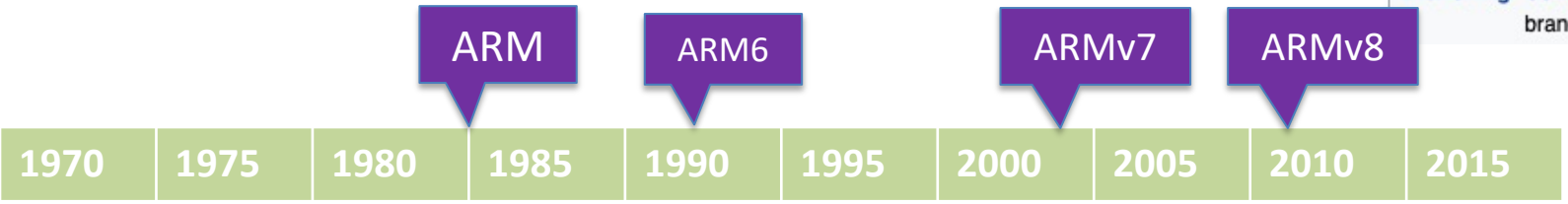
ARM architecture

From Wikipedia, the free encyclopedia

Arm architectures

The Arm logo

Designer	Arm Holdings
Bits	32-bit, 64-bit
Introduced	1985; 35 years ago
Design	RISC
Type	Register-Register
Branching	Condition code, compare and branch



Arm (previously officially written all caps as **ARM** and usually written as such today), previously **Advanced RISC Machine**, originally **Acorn RISC Machine**, is a family of reduced instruction set computing (RISC) architectures for computer processors, configured for various environments. Arm Holdings develops the architecture and licenses it to other companies, who design their own products that implement one of those architectures—including systems-on-chips (SoC) and systems-on-modules (SoM) that incorporate memory, interfaces, radios, etc. It also designs cores that implement this instruction set and licenses these designs to a number of companies that incorporate those core designs into their own products.

Processors that have a RISC architecture typically require fewer transistors than those with a complex instruction set computing (CISC) architecture (such as the x86 processors found in most personal computers), which improves cost, power consumption, and heat dissipation. These characteristics are desirable for light, portable, battery-powered devices—including smartphones, laptops and tablet computers, and other embedded systems[3][4][5]—but are also useful for servers and desktops to some degree. For supercomputers, which consume large amounts of electricity, Arm is also a power-efficient solution.[6]

CISC vs RISC:

Complex/Reduced Instruction Set Architecture

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

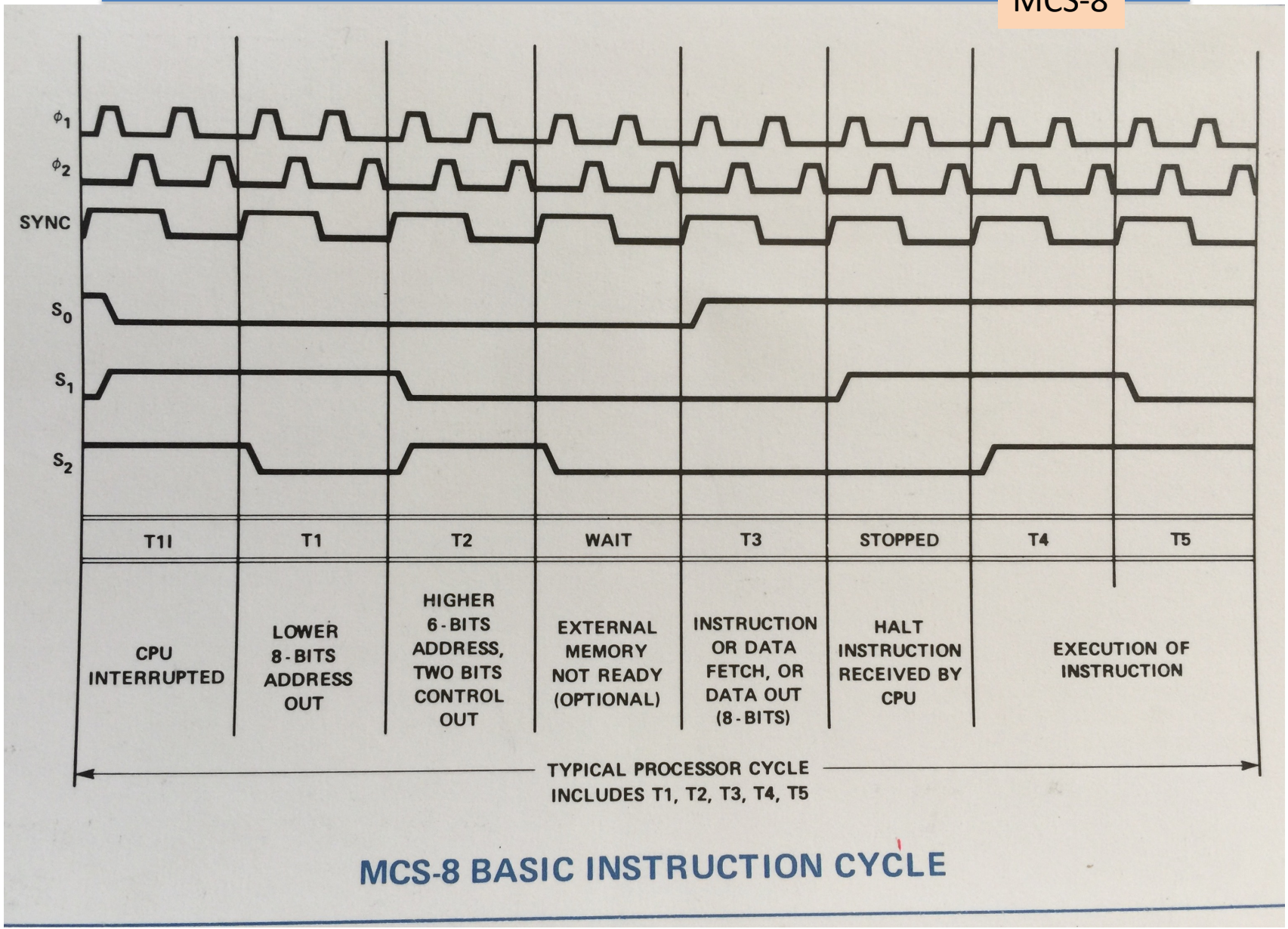
- Microprocessor History
  - 1971-85: **CISC** (8/16-bit)
    - Intel i4004 (4-bit)
    - Intel i8008 (8-bit) → i8080 → i8085, Z80 → i8086 (16-bit) → "x86"
    - Motorola 6800 (8-bit) → 6502 → 68000 (16-bit)
    - IBM PC used i8088 (8/16-bit) in 1981 → i80n86 ("x86") → *Pentiums* (now RISC)
  - 1985-2000: **RISC** – (32/64-bit)
    - SPARC* (UC Berkeley→ Sun/Oracle)
    - MIPS* (Stanford)
    - PowerPC (Motorola/IBM)
    - AMD 29K
    - Intel i960
    - ARM*

*still exist

CISC Instruction Cycle

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

MCS-8



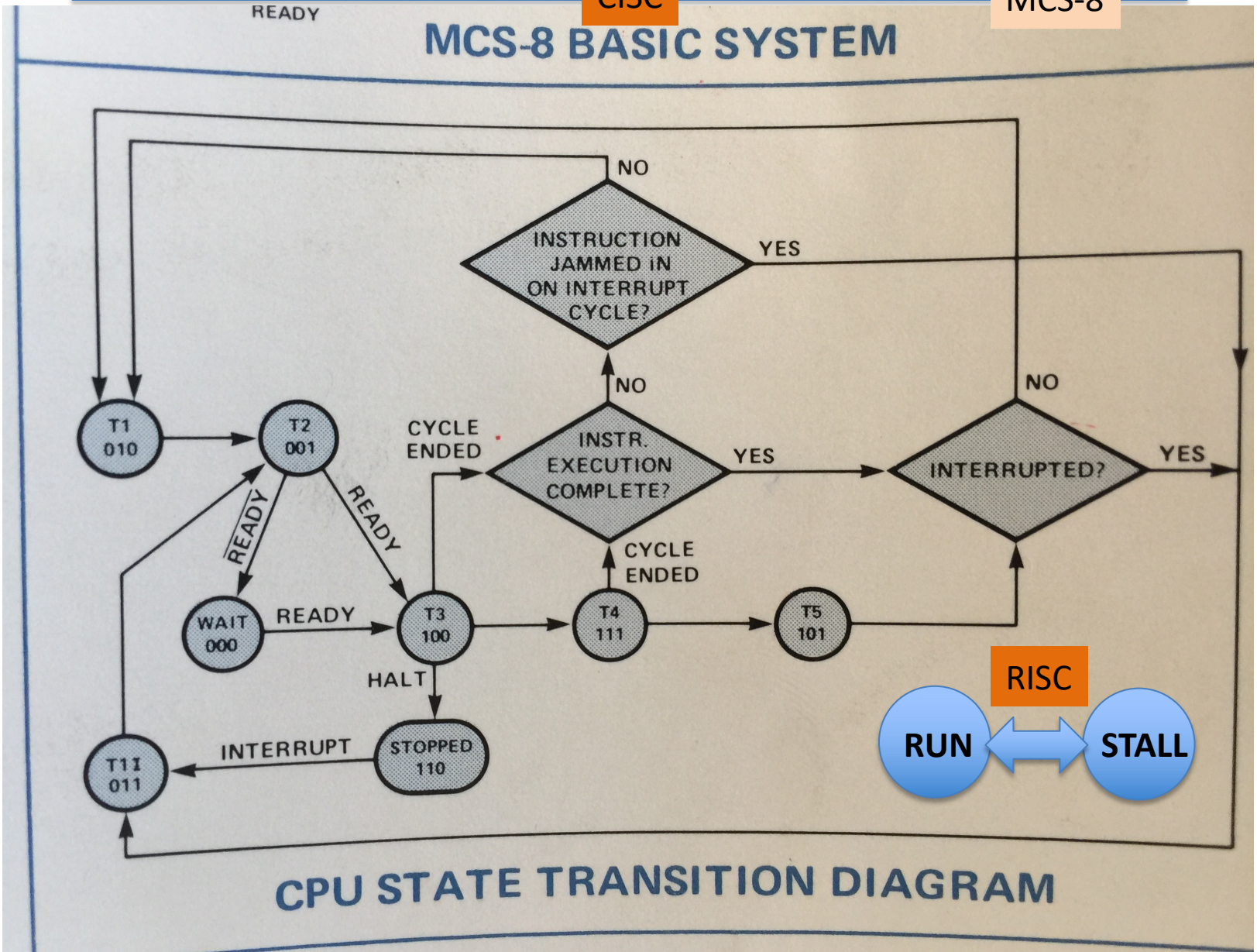
MCS-8 BASIC INSTRUCTION CYCLE

CISC State Diagram

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

CISC

MCS-8



RISC:

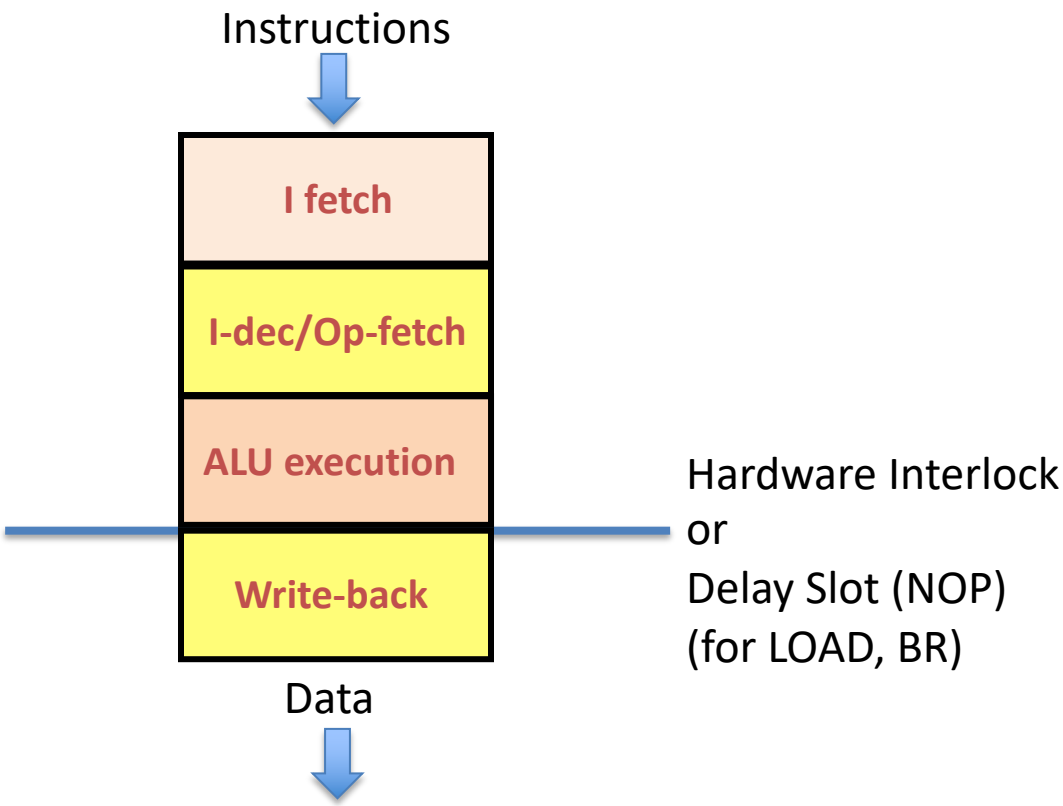
Reduced Instruction Set Architecture

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

- Key Architecture of RISC
  - Reduced ISA: small set of instructions (minor)
  - Fast execution: single cycle only
  - Reduced impact of memory
    - No *microprogram* (key change)
      - Instructions scale to *vertical microinstructions* (single-cycle)
      - eliminates ~30% chip area
    - LOAD-STORE (only) memory references
    - Full general register sets
    - Cache memory
      - On-chip
      - Multi-level
      - *Harvard* architecture – separate I and D
  - Pipelining
    - 4 or 5 stages
    - Interlocks
      - Hardware (SPARC, 29K)
      - Software (MIPS): compiler manages pipeline scheduling

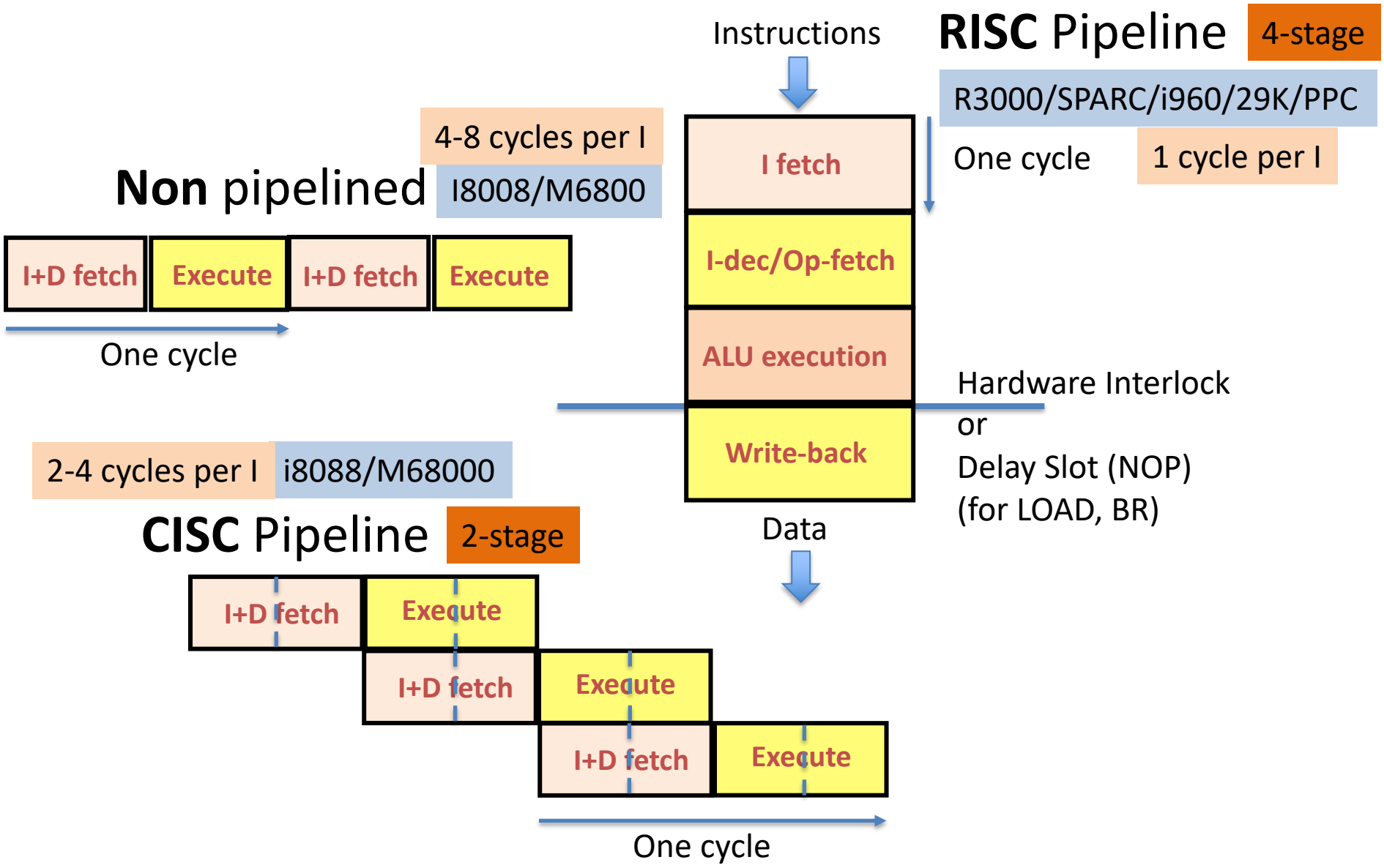
RISC Pipelines

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



CISC/RISC Pipelines

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Microcontrollers (MCU)

1st MCU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Quora

Tom Crosley
B.S. in Electrical Engineering, Iowa State University · June 30

Which is the first microcontroller?

The very first microcontroller was invented by Texas Instruments in 1971, and was called the TMS1802NC. It was used at TI internally in its calculator products between 1972 and 1974.

TI
TMS

The first commercially available microcontroller was the TMS 1000, also from Texas Instruments. It was released in 1974. It combined read-only memory, read/write memory, processor and clock on one chip and was targeted at embedded systems. The TMS 1000 was used in Texas Instruments' own Speak & Spell educational toy.

Intel

In response to this, Intel came out with the 8048 family in 1976. It was mostly replaced by the 8051 in 1980, which became on of the most widely used microcontrollers.

Note this was before user programmable memory became available. These early microcontrollers all used masked read-only memory (ROM), which meant the program was developed using an emulator, and when declared ready for production, a binary file would be sent to TI or Intel, and chips would be manufactured with the program already burned into them. This resulted in long lead times, and a disaster if a bug was found after the chips were programmed.

i8051 MCU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Intel 8051

From Wikipedia, the free encyclopedia

MCS-51 1980

The **Intel MCS-51** (commonly termed **8051**) is a single chip microcontroller (MCU) series developed by Intel in 1980 for use in embedded systems. The architect of the Intel MCS-51 instruction set was John H. Wharton.[1][2] Intel's original versions were popular in the 1980s and early 1990s and enhanced binary compatible derivatives remain popular today. It is an example of a complex instruction set computer, and has separate memory spaces for program instructions and data.

Intel P8051 microcontroller

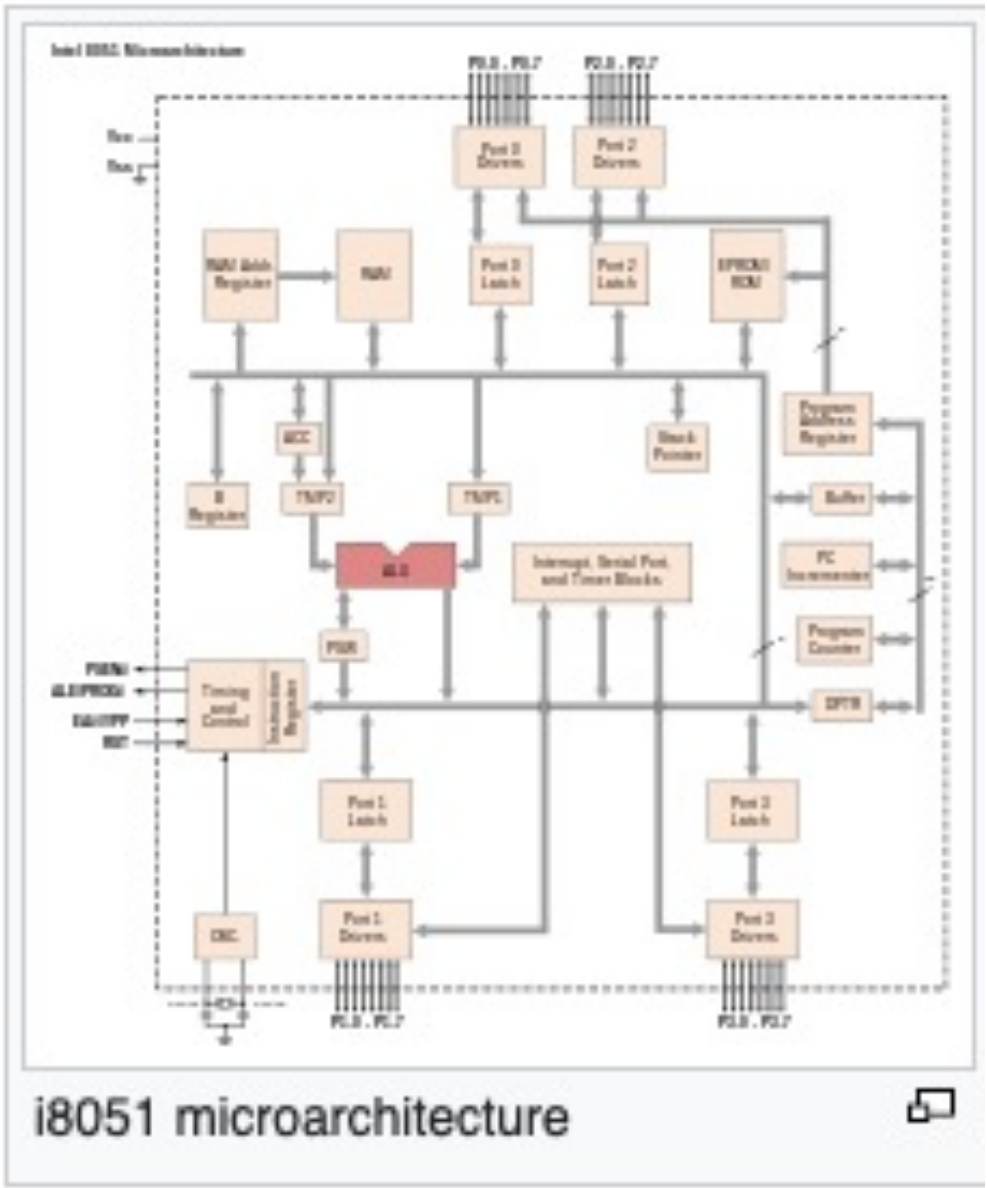
Intel's original MCS-51 family was developed using N-type metal-oxide-semiconductor (NMOS) technology like its predecessor Intel MCS-48, but later versions, identified by a letter C in their name (e.g., 80C51) use complementary metal–oxide–semiconductor (CMOS) technology and consume less power than their NMOS predecessors. This made them more suitable for battery-powered devices.

The family was continued in 1996 with the enhanced 8-bit MCS-151 and the 8/16/32-bit MCS-251 family of binary compatible microcontrollers.[3] While Intel no longer manufactures the MCS-51, MCS-151 and MCS-251 family, enhanced binary compatible derivatives made by numerous vendors remain popular today. Some derivatives integrate a digital signal processor (DSP). Beyond these physical devices, several companies also offer MCS-51 derivatives as IP cores for use in field programmable gate array (FPGA) or application specific integrated circuit (ASIC) designs.

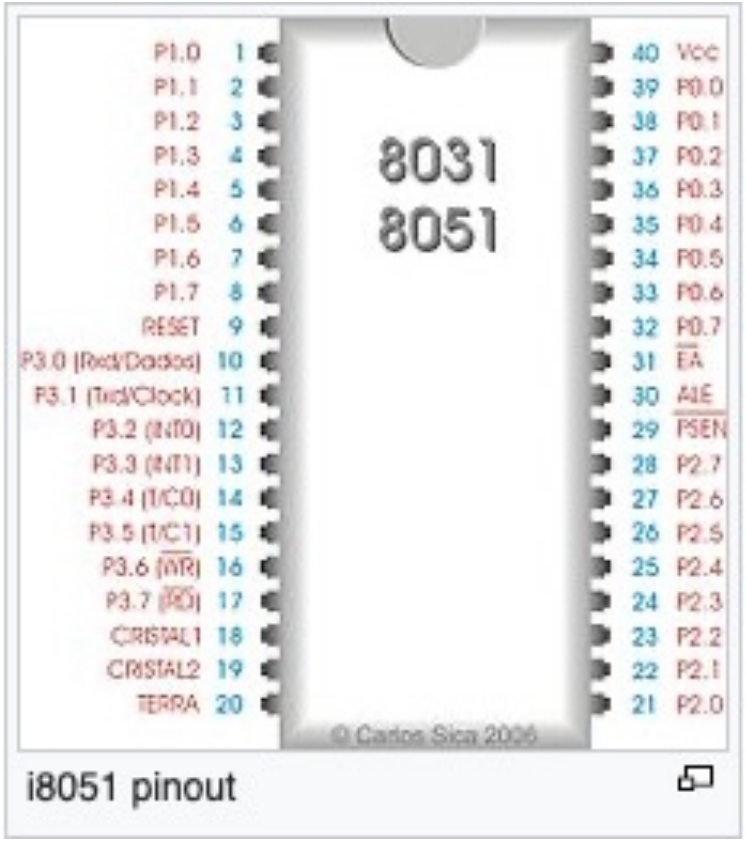
Intel P8051 microcontroller

i8051 MCU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



i8051 microarchitecture



i8051 pinout

Intel i8051 MCU 2nd Sources

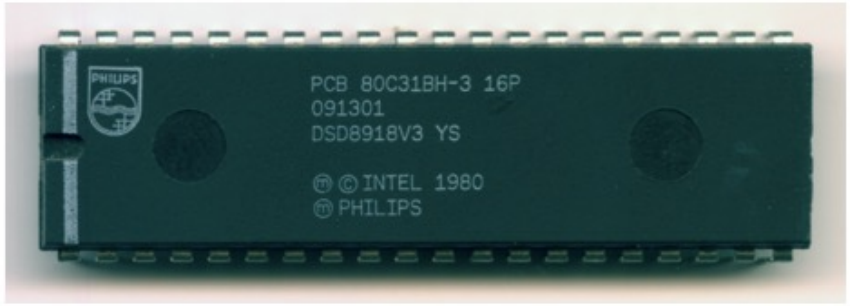
Jeff Drobman
©2016-23

Intel MCS-51 second sources

AMD D87C51

MHS S-80C31

OKI M80C31



Philips PCB80C31

Signetics SCN8031

Temic TS80C32

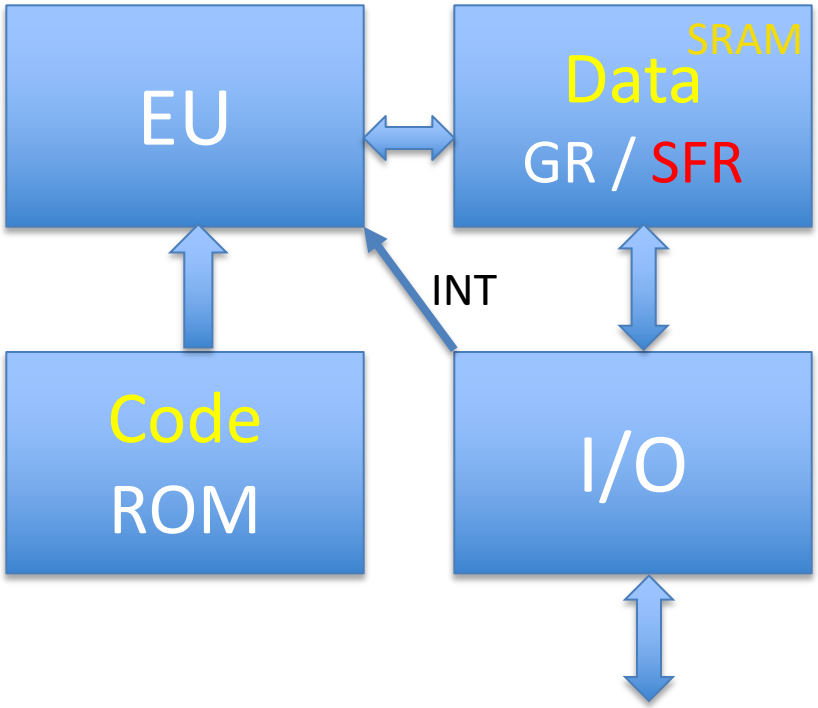
© 2022 Dr Jeff Drobman drjeff@drjeffsoftware.com

MCU Block Diagram

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

8/16/32-bit

BASIC MODEL



All on one cheap chip

- **No Cache**
- **No External RAM**

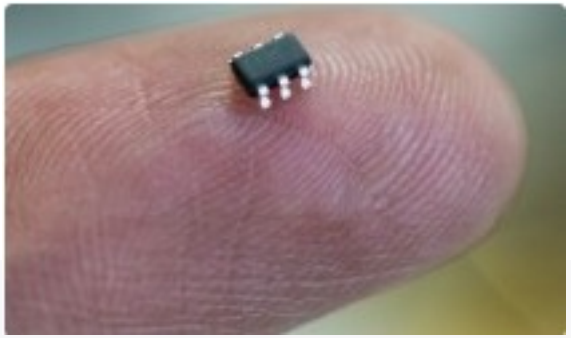
MCU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Meanwhile the number of microcontrollers estimated to be shipped in 2019 was estimated at around 27 billion, twelve times as many as the total number of microprocessors. As of 2017, the split was 40% for 32-bit, 33% 8-bit, and 24% 16-bit.

MCU = 12x MPU

So it can be estimated there were somewhere around nine billion 8-bit microcontrollers shipped in 2019. They are predominantly used in embedded systems that have a specific task, such as a small (air fryer, microwave oven) or large (washing machine) appliance; automobile cruise control; intelligent thermostat; etc.



Jeff Drobman

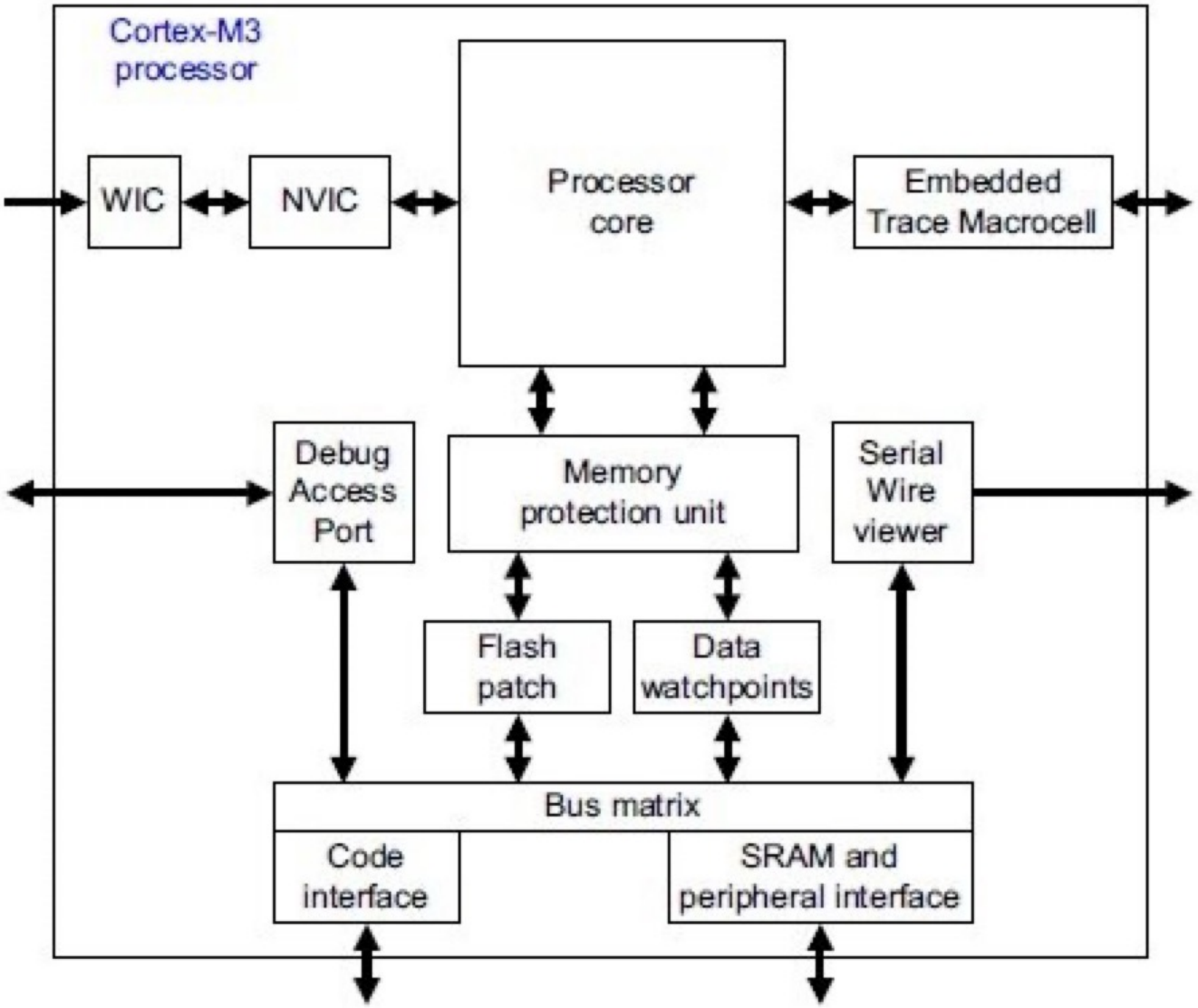
Just now

as of 5 years ago (when I last checked), the i8051 was still popular along with the PIC16 and 18 (16-bit). many models sold at <$1. Atmel's AVR is a popular microcontroller family that is customizable.

ARM Cortex M3

Quora

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

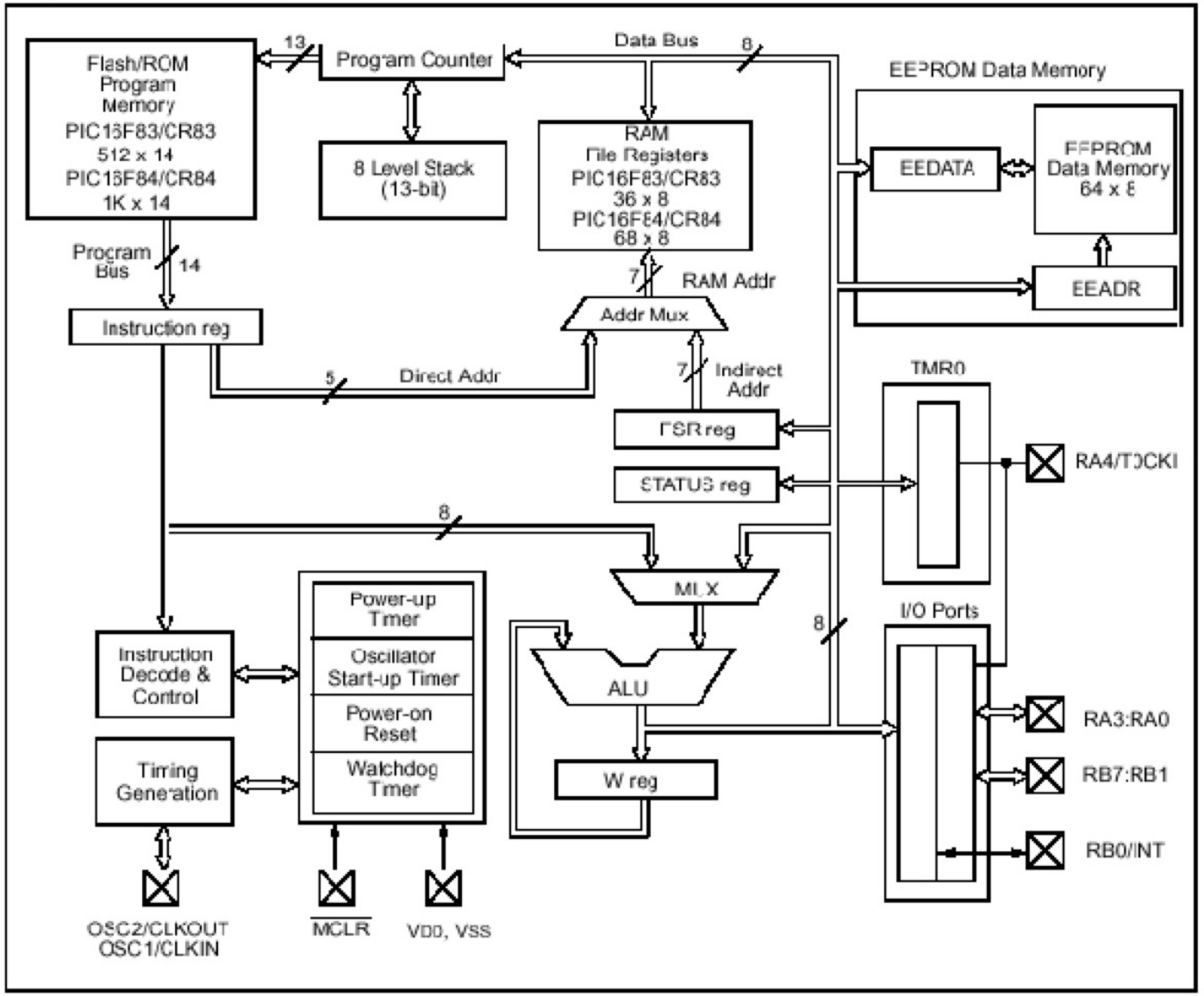


PIC 16F MCU

Quora

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

FIGURE 3-1: PIC16F8X BLOCK DIAGRAM



Embedded Control

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Microprocessors
For
COMPUTING

- All 32/64-bit CPUs
- Large *data processing* applications
  - Employee records
  - Accounting
  - Payroll
- Operating systems (OS)
- "Apps" (applications)
  - PC/Mac
  - Mobile (phones, tablets)
  - Web apps
  - Cloud apps (SaaS)

Focus is ***Memory*** for large Data Files

Large DRAM, Disk, Flash

Microcontrollers
For
CONTROL

- *Real-time*
- *All-in-one*

- Small *embedded control* applications (8-bit MCU)
  - Appliances
  - Disk controllers
  - Remote controllers
  - Garage/gate openers
- Tiny
- Low power
- Low cost
- Medium *embedded control* (16-bit MCU)
  - User devices (iPods, phones, etc.)
  - Car/Airplane engine control
  - Car/Airplane braking & safety
  - Car transmission control
  - Home Automation (HAN)
- Large *embedded control* (32/64-bit MCU)
  - Car/Airplane entertainment
  - Car/Airplane navigation, systems management
  - Printers (MF)
  - Communications gear (WiFi, cable TV boxes)

Focus is ***I/O*** – *Interrupts*

Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Microprocessor Timeline (Exhaustive)

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wikipedia

Date	Name	Developer	Max clock (first version)	Word size (bits)	Process	Chips[5]	Transistors	MOSFET
1971	4004	Intel	740 kHz	4	10 µm	1	2,250	pMOS
1972	PPS-25	Fairchild	400 kHz	4		2		pMOS
1972	µPD700	NEC		4		1		
1972	8008	Intel	500 kHz	8	10 µm	1	3,500	pMOS
1972	PPS-4	Rockwell	200 kHz	4		1		pMOS
1973	µCOM-4	NEC	2 MHz	4	7.5 µm	1	2,500	NMOS
1973	TLCS-12	Toshiba	1 MHz	12	6 µm	1	2,800 silicon gates	pMOS
1973	Mini-D	Burroughs	1 MHz	8		1		pMOS
1974	IMP-8	National	715 kHz	8		3		pMOS
1974	8080	Intel	2 MHz	8	6 µm	1	6,000	NMOS
1974	µCOM-8	NEC	2 MHz	8		1		NMOS
1974	5065	Mostek	1.4 MHz	8		1		pMOS
1974	µCOM-16	NEC	2 MHz	16		2		NMOS
1974	IMP-4	National	500 kHz	4		3		pMOS
1974	4040	Intel	740 kHz	4	10 µm	1	3,000	pMOS
1974	6800	Motorola	1 MHz	8	-	1	4,100	NMOS
1974	TMS 1000	Texas Instruments	400 kHz	4	8 µm	1	8,000	
1974	PACE	National		16		1		pMOS
1974	ISP-8A/500 (SC/MP)	National	1 MHz	8		1		pMOS
1975	6100	Intersil	4 MHz	12	-	1	4,000	CMOS
1975	TLCS-12A	Toshiba	1.2 MHz	12	-	1		pMOS
1975	2650	Signetics	1.2 MHz	8		1		NMOS
1975	PPS-8	Rockwell	256 kHz	8		1		pMOS

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wikipedia

1975	F-8	Fairchild	2 MHz	8		1		NMOS
1975	CDP 1801	RCA	2 MHz	8	5 µm	2	5,000	CMOS
1975	6502	MOS Technology	1 MHz	8	-	1	3,510	NMOS (dynamic)
1975	IMP-16	National	715 kHz	16		5		pMOS
1975	PFL-16A (MN 1610)	Panafacom	2 MHz	16	-	1		NMOS
1975	BPC	Hewlett Packard	10 MHz	16	-	1	6,000 (+ ROM)	NMOS
1975	MCP-1600	Western Digital	3.3 MHz	16	-	3		PMOS
1975	CP1600	General Instrument	3.3 MHz	16		1		NMOS
1976	CDP 1802	RCA	6.4 MHz	8		1		CMOS
1976	Z-80	Zilog	2.5 MHz	8	4 µm	1	8,500	NMOS
1976	TMS9900	Texas Instruments	3.3 MHz	16	-	1	8,000	
1976	8x300	Signetics	8 MHz	8		1		Bipolar
1976	WD16	Western Digital	3.3 MHz	16		5		PMOS
1977	Bellmac-8 (WE212)	Bell Labs	2.0 MHz	8	5 µm	1	7,000	CMOS
1977	8085	Intel	3.0 MHz	8	3 µm	1	6,500	
1977	MC14500B	Motorola	1.0 MHz	1		1		CMOS
1978	6809	Motorola	1 MHz	8	5 µm	1	9,000	
1978	8086	Intel	5 MHz	16	3 µm	1	29,000	
1978	6801	Motorola	-	8	5 µm	1	35,000	
1979	Z8000	Zilog	-	16	-	1	17,500	
1979	8088	Intel	5 MHz	8/16[b]	3 µm	1	29,000	NMOS (HMOS)
1979	68000	Motorola	8 MHz	16/32[c]	3.5 µm	1	68,000	NMOS (HMOS)

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wikipedia

Date	Name	Developer	Clock	Word size (bits)	Process	Transistors
1980	16032	National Semiconductor	-	16/32	-	60,000
1981	6120	Harris Corporation	10 MHz	12	-	20,000 (CMOS)[36]
1981	ROMP	IBM	10 MHz	32	2 µm	45,000
1981	T-11	DEC	2.5 MHz	16	5 µm	17,000 (NMOS)
1982	RISC-I[37]	UC Berkeley	1 MHz	-	5 µm	44,420 (NMOS)
1982	FOCUS	Hewlett Packard	18 MHz	32	1.5 µm	450,000
1982	80186	Intel	6 MHz	16	-	55,000
1987	80C186	Intel	10 MHz	16	-	56,000 (CMOS)
1982	80188	Intel	8 MHz	8/16	-	29,000
1982	80286	Intel	6 MHz	16	1.5 µm	134,000
1983	RISC-II	UC Berkeley	3 MHz	-	3 µm	40,760 (NMOS)
1983	MIPS[38]	Stanford University	2 MHz	32	3 µm	25,000
1983	65816	Western Design Center	-	16	-	-
1984	68020	Motorola	16 MHz	32	2 µm	190,000
1984	NS32032	National Semiconductor	-	32	-	70,000
1984	V20	NEC	5 MHz	8/16	-	63,000
1985	80386	Intel	16–40 MHz	32	1.5 µm	275,000
1985	MicroVax II 78032	DEC	5 MHz	32	3.0 µm	125,000
1985	R2000	MIPS	8 MHz	32	2 µm	115,000
1985[39]	Novix NC4016	Harris Corporation	8 MHz	16	3 µm[40]	16,000[41]
1986	Z80000	Zilog	-	32	-	91,000
1986	SPARC MB86900	Fujitsu[42][43][44]	40 MHz	32	0.8 µm	800,000
1986	V60[45]	NEC	16 MHz	16/32	1.5 µm	375,000

©2016-23

Wikipedia

1986	SPARC MB86900	Fujitsu[42][43][44]	40 MHz	32	0.8 µm	800,000
1986	V60[45]	NEC	16 MHz	16/32	1.5 µm	375,000
1987	CVAX 78034	DEC	12.5 MHz	32	2.0 µm	134,000
1987	ARM2	Acorn	8 MHz	32	2 µm	25,000[46]
1987	Gmicro/200[47]	Hitachi	-	-	1 µm	730,000
1987	68030	Motorola	16 MHz	32	1.3 µm	273,000
1987	V70[45]	NEC	20 MHz	16/32	1.5 µm	385,000
1988	R3000	MIPS	25 MHz	32	1.2 µm	120,000
1988	80386SX	Intel	12–33 MHz	16/32	-	-
1988	i960	Intel	10 MHz	33/32	1.5 µm	250,000
1989	i960CA[48]	Intel	16–33 MHz	33/32	0.8 µm	600,000
1989	VAX DC520 "Rigel"	DEC	35 MHz	32	1.5 µm	320,000
1989	80486	Intel	25 MHz	32	1 µm	1,180,000
1989	i860	Intel	25 MHz	32	1 µm	1,000,000

Jeff Drobman
©2016-23

Wikipedia

1995	UltraSPARC	Sun	143–167 MHz	64	470 nm	5.2
1995	SPARC64	HAL Computer Systems	101–118 MHz	64	400 nm	-
1995	Pentium Pro	Intel	150–200 MHz	32	350 nm	5.5
1996	Alpha 21164A	DEC	400–500 MHz	64	350 nm	9.7
1996	K5	AMD	75–100 MHz	32	500 nm	4.3
1996	R10000	MTI	150–250 MHz	64	350 nm	6.7
1996	R5000	QED	180–250 MHz	-	350 nm	3.7
1996	SPARC64 II	HAL Computer Systems	141–161 MHz	64	350 nm	-
1996	PA-8000	Hewlett-Packard	160–180 MHz	64	500 nm	3.8
1996	POWER2 Super Chip (P2SC)	IBM	150 MHz	32	290 nm	15
1997	SH-4	Hitachi	200 MHz	-	200 nm[54]	10[55]
1997	RS64	IBM	125 MHz	64	? nm	?
1997	Pentium II	Intel	233–300 MHz	32	350 nm	7.5
1997	PowerPC 620	IBM, Motorola	120–150 MHz	64	350 nm	6.9
1997	UltraSPARC IIs	Sun	250–400 MHz	64	350 nm	5.4
1997	S/390 G4	IBM	370 MHz	32	500 nm	7.8
1997	PowerPC 750	IBM, Motorola	233–366 MHz	32	260 nm	6.35
1997	K6	AMD	166–233 MHz	32	350 nm	8.8
1998	RS64-II	IBM	262 MHz	64	350 nm	12.5
1998	Alpha 21264	DEC	450–600 MHz	64	350 nm	15.2
1998	MIPS R12000	SGI	270–400 MHz	64	250–180 nm	6.9
1998	RM7000	QED	250–300 MHz	-	250 nm	18
1998	SPARC64 III	HAL Computer Systems	250–330 MHz	64	240 nm	17.6
1998	S/390 G5	IBM	500 MHz	32	250 nm	25
1998	PA-8500	Hewlett Packard	300–440 MHz	64	250 nm	140

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wikipedia

Date	Name	Developer	Clock	Process	Transistors (millions)	Cores per die / Dies per module
2000	Athlon XP	AMD	1.33–1.73 GHz	180 nm	37.5	1 / 1
2000	Duron	AMD	550 MHz–1.3 GHz	180 nm	25	1 / 1
2000	RS64-IV	IBM	600–750 MHz	180 nm	44	1 / 2
2000	Pentium 4	Intel	1.3–2 GHz	180–130 nm	42	1 / 1
2000	SPARC64 IV	Fujitsu	450–810 MHz	130 nm	-	1 / 1
2000	z900	IBM	918 MHz	180 nm	47	1 / 12, 20
2001	MIPS R14000	SGI	500–600 MHz	130 nm	7.2	1 / 1
2001	POWER4	IBM	1.1–1.4 GHz	180–130 nm	174	2 / 1, 4
2001	UltraSPARC III	Sun	750–1200 MHz	130 nm	29	1 / 1
2001	Itanium	Intel	733–800 MHz	180 nm	25	1 / 1
2001	PowerPC 7450	Motorola	733–800 MHz	180–130 nm	33	1 / 1
2002	SPARC64 V	Fujitsu	1.1–1.35 GHz	130 nm	190	1 / 1
2002	Itanium 2	Intel	0.9–1 GHz	180 nm	410	1 / 1
2003	PowerPC 970	IBM	1.6–2.0 GHz	130–90 nm	52	1 / 1
2003	Pentium M	Intel	0.9–1.7 GHz	130–90 nm	77	1 / 1
2003	Opteron	AMD	1.4–2.4 GHz	130 nm	106	1 / 1
2004	POWER5	IBM	1.65–1.9 GHz	130–90 nm	276	2 / 1, 2, 4
2004	PowerPC BGL	IBM	700 MHz	130 nm	95	2 / 1
2005	Opteron "Athens"	AMD	1.6–3.0 GHz	90 nm	114	1 / 1
2005	Pentium D	Intel	2.8–3.2 GHz	90 nm	115	1 / 2
2005	Athlon 64 X2	AMD	2–2.4 GHz	90 nm	243	2 / 1
2005	PowerPC 970MP	IBM	1.2–2.5 GHz	90 nm	183	2 / 1
2005	UltraSPARC IV	Sun	1.05–1.35 GHz	130 nm	66	2 / 1

©2016-23

Wikipedia

2005	UltraSPARC T1	Sun	1–1.4 GHz	90 nm	300	8 / 1
2005	Xenon	IBM	3.2 GHz	90–45 nm	165	3 / 1
2006	Core Duo	Intel	1.1–2.33 GHz	90–65 nm	151	2 / 1
2006	Core 2	Intel	1.06–2.67 GHz	65–45 nm	291	2 / 1, 2
2006	Cell/B.E.	IBM, Sony, Toshiba	3.2–4.6 GHz	90–45 nm	241	1+8 / 1
2006	Itanium "Montecito"	Intel	1.4–1.6 GHz	90 nm	1720	2 / 1
2007	POWER6	IBM	3.5–4.7 GHz	65 nm	790	2 / 1
2007	SPARC64 VI	Fujitsu	2.15–2.4 GHz	90 nm	543	2 / 1
2007	UltraSPARC T2	Sun	1–1.4 GHz	65 nm	503	8 / 1
2007	TILE64	Tilera	600–900 MHz	90–45 nm	?	64 / 1
2007	Opteron "Barcelona"	AMD	1.8–3.2 GHz	65 nm	463	4 / 1
2007	PowerPC BGP	IBM	850 MHz	90 nm	208	4 / 1
2008	Phenom	AMD	1.8–2.6 GHz	65 nm	450	2, 3, 4 / 1
2008	z10	IBM	4.4 GHz	65 nm	993	4 / 7
2008	PowerXCell 8i	IBM	2.8–4.0 GHz	65 nm	250	1+8 / 1
2008	SPARC64 VII	Fujitsu	2.4–2.88 GHz	65 nm	600	4 / 1
2008	Atom	Intel	0.8–1.6 GHz	65–45 nm	47	1 / 1
2008	Core i7	Intel	2.66–3.2 GHz	45–32 nm	730	2, 4, 6 / 1
2008	TILEPro64	Tilera	600–866 MHz	90–45 nm	?	64 / 1
2008	Opteron "Shanghai"	AMD	2.3–2.9 GHz	45 nm	751	4 / 1
2009	Phenom II	AMD	2.5–3.2 GHz	45 nm	758	2, 3, 4, 6 / 1
2009	Opteron "Istanbul"	AMD	2.2–2.8 GHz	45 nm	904	6 / 1

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wikipedia

Date	Name	Developer	Clock	Process	Transistors (millions)	Cores per die / Dies per module	threads per core
2010	POWER7	IBM	3–4.14 GHz	45 nm	1200	4, 6, 8 / 1, 4	4
2010	Itanium "Tukwila"	Intel	2 GHz	65 nm	2000	2, 4 / 1	2
2010	Opteron "Magny-cours"	AMD	1.7–2.4 GHz	45 nm	1810	4, 6 / 2	1
2010	Xeon "Nehalem-EX"	Intel	1.73–2.66 GHz	45 nm	2300	4, 6, 8 / 1	2
2010	z196	IBM	3.8–5.2 GHz	45 nm	1400	4 / 1, 6	1
2010	SPARC T3	Sun	1.6 GHz	45 nm	2000	16 / 1	8
2010	SPARC64 VII+	Fujitsu	2.66–3.0 GHz	45 nm	?	4 / 1	2
2010	Intel "Westmere"	Intel	1.86–3.33 GHz	32 nm	1170	4–6 / 1	2
2011	Intel "Sandy Bridge"	Intel	1.6–3.4 GHz	32 nm	995[58]	2, 4 / 1	(1,) 2
2011	AMD Llano	AMD	1.0–1.6 GHz	40 nm	380[59]	1, 2 / 1	1
2011	Xeon E7	Intel	1.73–2.67 GHz	32 nm	2600	4, 6, 8, 10 / 1	1–2
2011	Power ISA BGQ	IBM	1.6 GHz	45 nm	1470	18 / 1	4
2011	SPARC64 VIIIfx	Fujitsu	2.0 GHz	45 nm	760	8 / 1	2
2011	FX "Bulldozer" Interlagos	AMD	3.1–3.6 GHz	32 nm	1200[60]	4–8 / 2	1
2011	SPARC T4	Oracle	2.8–3 GHz	40 nm	855	8 / 1	8
2012	SPARC64 IXfx	Fujitsu	1.848 GHz	40 nm	1870	16 / 1	2
2012	zEC12	IBM	5.5 GHz	32 nm	2750	6 / 6	1
2012	POWER7+	IBM	3.1–5.3 GHz	32 nm	2100	8 / 1, 2	4
2012	Itanium "Poulson"	Intel	1.73–2.53 GHz	32 nm	3100	8 / 1	2
2013	Intel "Haswell"	Intel	1.9–4.4 GHz	22 nm	1400	4 / 1	2
2013	SPARC64 X	Fujitsu	2.8–3 GHz	28 nm	2950	16 / 1	2
2013	SPARC T5	Oracle	3.6 GHz	28 nm	1500	16 / 1	8
2014	POWER8	IBM	2.5–5 GHz	22 nm	4200	6, 12 / 1, 2	8

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

2014	POWER8	IBM	2.5–5 GHz	22 nm	4200	6, 12 / 1, 2	8
2014	Intel "Broadwell"	Intel	1.8-4 GHz	14 nm	1900	2, 4, 6, 8, 12, 16 / 1, 2, 4	2
2015	z13	IBM	5 GHz	22 nm	3990	8 / 1	2
2015	A8-7670K	AMD	3.6 GHz	28 nm	2410	4 / 1	1
2017	Zen	AMD	3.2–4.1 GHz	14 nm	4800	8, 16, 32 / 1, 2, 4	2
2017	z14	IBM	5.2 GHz	14 nm	6100	10 / 1	2
2017	POWER9	IBM	4 GHz	14 nm	8000	12, 24 / 1	4, 8
2017	SPARC M8[61]	Oracle	5 GHz	20 nm	~10,000[62]	32	8
2018	Intel "Cannon Lake"	Intel	2.2-3.2 GHz	10 nm	?	2 / 1	2
2018	Zen+	AMD	2.8-3.7 GHz	12 nm	4800	2, 4, 6, 8, 12, 16, 24, 32 / 1, 2, 4	1, 2
2019	Zen 2	AMD	2-4.7 GHz	7 nm	3900	6, 8, 12, 16, 24, 32, 64 / 1, 2, 4	2
2019	z15	IBM	5.2 GHz	14 nm	9200	12 / 1	2

2020s [edit]

Date	Name	Developer	Clock	Process	Transistors (millions)	Cores per die / Dies per module	threads per core
2020	Zen 3	AMD	3.4–4.9 GHz	7 nm	?	6, 8, 12, 16 /	2
2020	M1	Apple	3.2 GHz	5 nm	16000	8	1
2021	M1 Max	Apple	3.2 GHz	5 nm	57000	10	1

Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

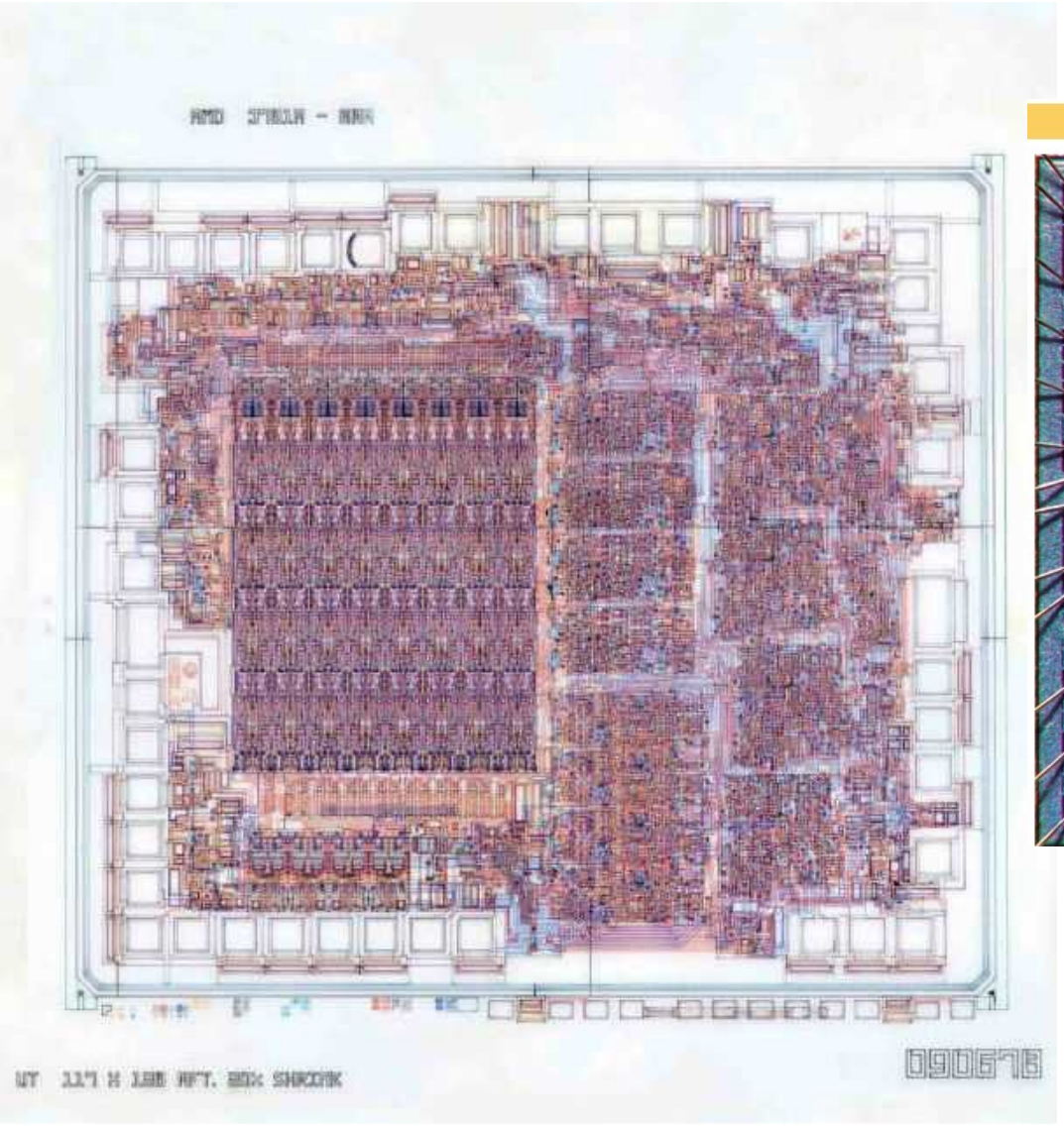
Early RISC Microprocessors

- AMD 29K
- Intel i960, XScale
- MIPS R2/3/4000

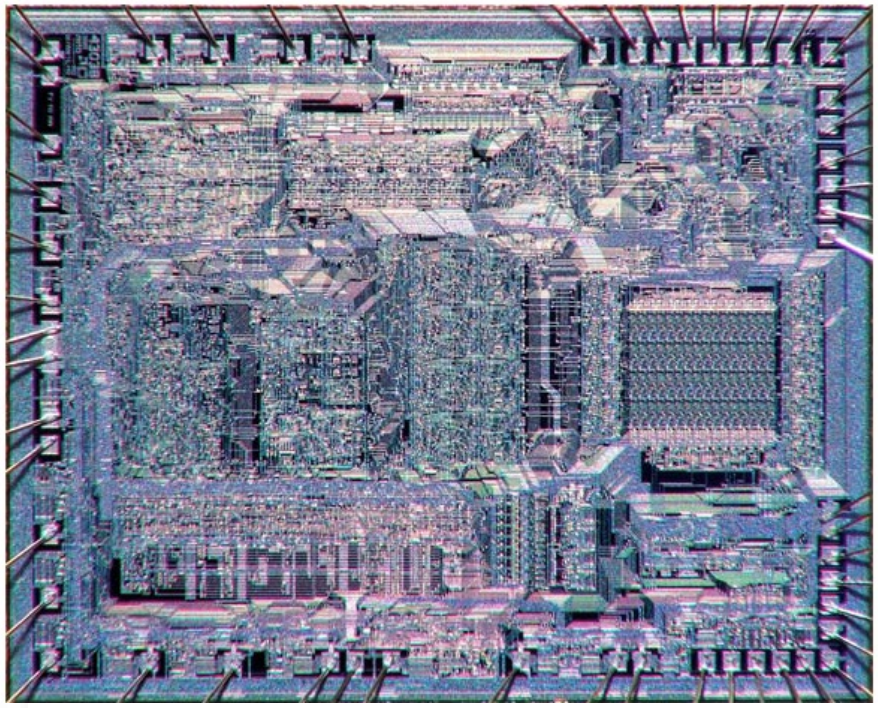
Am2901/3 4-bit MPU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Bit-slice 1975-85



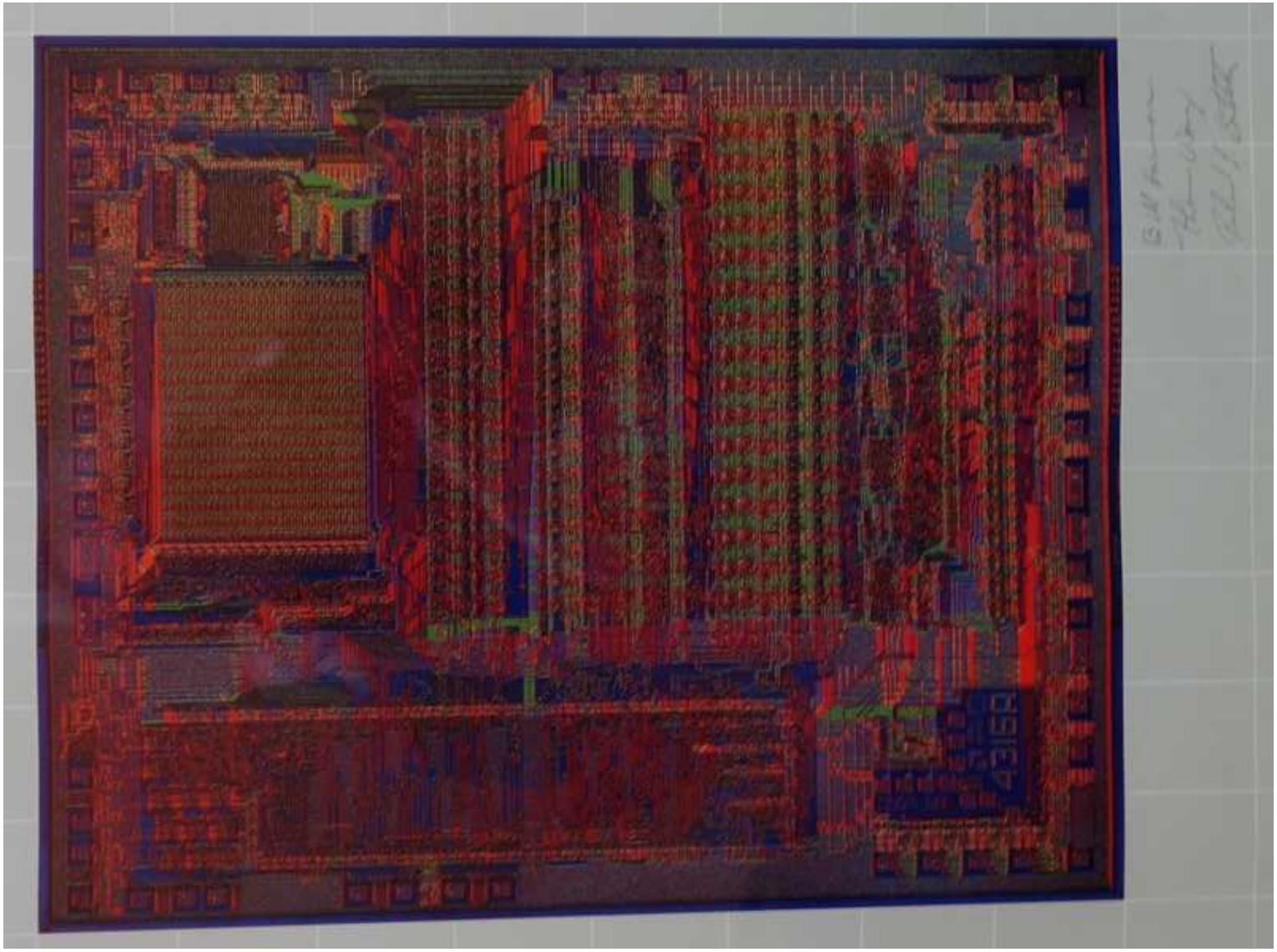
AMD AM2903ADC



Am29116 16-bit MPU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Bit-slice 1985-88



Am29K

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

AMD Am29000

Berkeley **RISC** (Patterson)

From Wikipedia, the free encyclopedia

The **AMD Am29000**, commonly shortened to **29k**, is a family of 32-bit RISC microprocessors and microcontrollers developed and fabricated by Advanced Micro Devices (AMD). Based on the seminal Berkeley RISC, the 29k added a number of significant improvements. They were, for a time, the most popular RISC chips on the market, widely used in laser printers from a variety of manufacturers.

Several versions were introduced during the period from 1988 to 1995, beginning with the 29000. The final model, the **29050**, was the first superscalar version, retiring up to four instructions per cycle, and also including a greatly improved floating point unit (FPU).

In late 1995 AMD dropped development of the 29k because the design team was transferred to support the PC side of the business. What remained of AMD's embedded business was realigned towards the embedded 186 family of 80186 derivatives. The majority of AMD's resources were then concentrated on their high-performance, desktop x86 clones, using many of the ideas and individual parts of the latest 29k to produce the AMD K5.

The 29000 evolved from the same Berkeley RISC design that also led to the Sun SPARC and Intel i960.

One design element used in all of the Berkeley-derived designs is the concept of register windows, a technique used to speed up procedure calls significantly. The idea is to use a large set of registers as a stack, loading local data into a set of registers

Register Windows

Am29K

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Register Windows

Design [edit]

AMD 29030.

The 29000 evolved from the same Berkeley RISC design that also led to the Sun SPARC and Intel i960.

One design element used in all of the Berkeley-derived designs is the concept of register windows, a technique used to speed up procedure calls significantly. The idea is to use a large set of registers as a stack, loading local data into a set of registers during a call, and marking them "dead" when the procedure returns. Values being returned from the routines would be placed in the "global page", the top eight registers in the SPARC (for instance). The competing early RISC design from Stanford University, the Stanford MIPS, also looked at this concept but decided that improved compilers could make more efficient use of general purpose registers than a hard-wired window.

In the original Berkeley design, SPARC, and i960, the windows were fixed in size. A routine using only one local variable would still use up eight registers on the SPARC, wasting this expensive resource. It was here that the 29000 differed from these earlier designs, using a variable window size. In this example only two registers would be used, one for the local variable, another for the return address. It also added more registers, including the same 128 registers for the procedure stack, but adding another 64 for global access. In comparison, the SPARC had 128 registers in total, and the global set was a standard window of eight. This change resulted in much better register use in the 29000 under a wide variety of workloads.

The 29000 also extended the register window stack with an in-memory (and in theory, in-cache) stack. When the window filled the calls would be pushed off the end of the register stack into memory, restored as required when the routine returned. Generally, the 29000's register usage was considerably more advanced than competing designs based on the Berkeley concepts.

Another difference with the Berkeley design is that the 29000 included no special-purpose condition code register. Any register could be used for this purpose, allowing the conditions to be easily saved at the expense of complicating some code. An instruction prefetch buffer was used that stored up to 16 instructions, used to improve performance during branches—the 29000 did not include any branch prediction system so there was a delay if a branch was taken (nor was it originally superscalar, so it could not "do both sides" as is common in some designs). The buffer mitigated this by storing four instructions from the other side of the branch, which could be run instantly while the buffer was re-filled with new instructions from memory.

Condition Codes = Flags

Am29K

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

AMD 29000 Microprocessor

AMD 29030.

AMD 29040

1988

1991-2

1994-5

Versions [edit]

The first 29000 was released in 1988, including a built-in MMU but floating point support was offloaded to the **29027** FPU. Units with failed MMU's or BTC's were sold as the **29005**.

Am29K

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Versions [edit]

The first 29000 was released in 1988, including a built-in MMU but floating point support was offloaded to the **29027** FPU. Units with failed MMU's or BTC's were sold as the **29005**.

The last general-purpose version was the **29050**. This was a significant upgrade to the original design, the first superscalar version which could execute instructions out-of-order and speculatively. Up to six instructions could be worked on at the same time through various pipeline stages, and four could be retired at any cycle. The 29050 also included a significantly improved FPU. The 29050 was initially available with clock rates of 25, 50, and 75 MHz. A 100 MHz version was introduced later.[1]

Several portions of the 29050 design were used as the basis for the K5 series of x86-compatible processors. The FPU adder and multiplier were carried over with some layout changes, a nanocode engine was added to the FPU to accommodate the complex instructions found in x86 but not on the 29050, while the rest of the core design was used along with complex microcode to translate x86 instructions to 29k-like 'uops' on the fly.

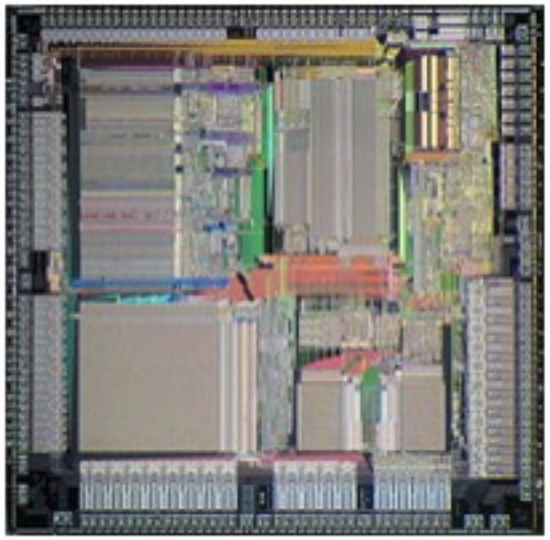
The Honeywell 29KII is a cpu based on the AMD 29050, and it was extensively used in real-time avionics.

29050 → K5 (x86 Pentium)

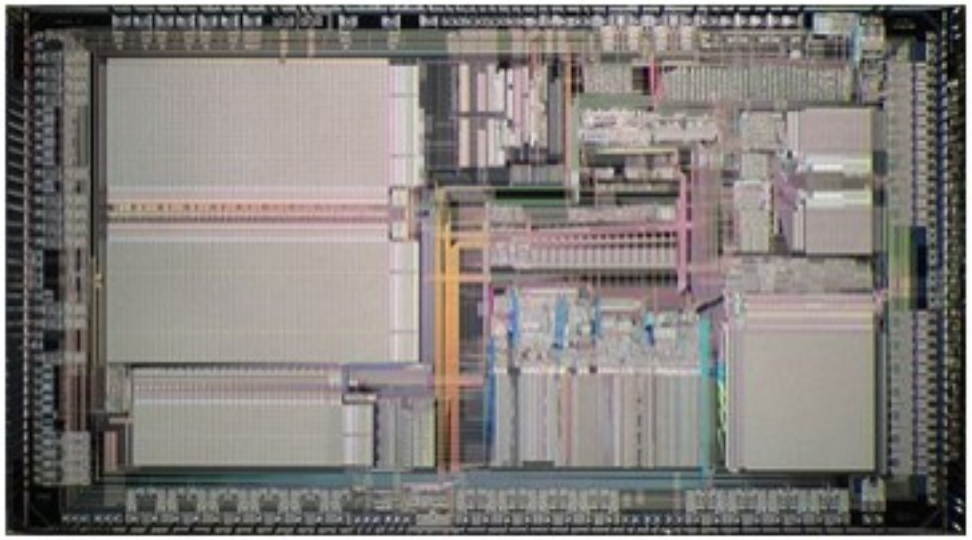
Am29K

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

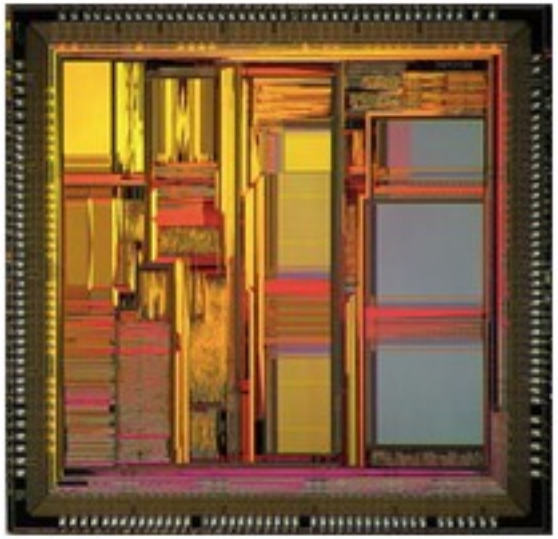
Die photos



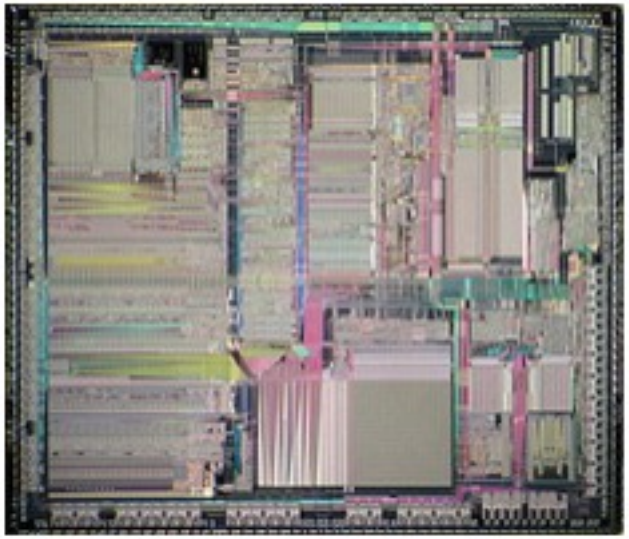
Am29000



Am29030



Am29040



Am29050

iAPX 432

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

The **iAPX 432** is a discontinued computer architecture introduced in 1981. It was Intel's first 32-bit processor design. The main processor of the architecture, the *general data processor*, is implemented as a set of two separate integrated circuits, due to technical limitations at the time. Although some es

Forerunner of **i960**

i960

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Intel i960

From Wikipedia, the free encyclopedia

Intel's **i960** (or **80960**) was a RISC-based microprocessor design that became popular during the early 1990s as an embedded microcontroller. It became a best-selling CPU in that segment, along with the competing AMD 29000.[2] In spite of its success, Intel stopped marketing the i960 in the late 1990s, as a result of a settlement with DEC whereby Intel received the rights to produce the StrongARM CPU. The processor continues to be used for a few military applications.

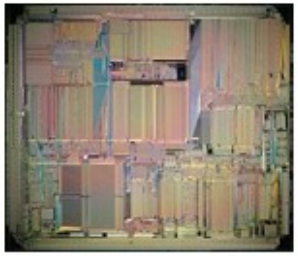
Intel i960HA microprocessor

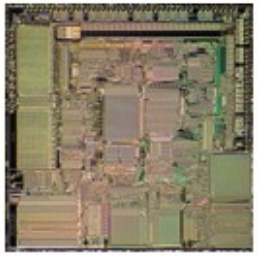
General Info	
Launched	1984
Discontinued	2007[1]
Common manufacturer(s)	Intel
Performance	
Max. CPU clock rate	10 MHz to 100 MHz

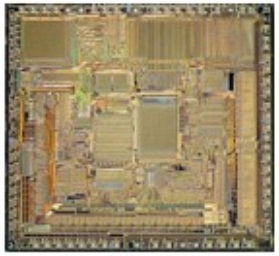
i960

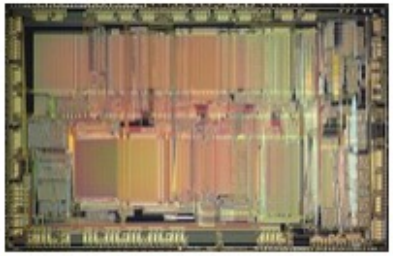
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

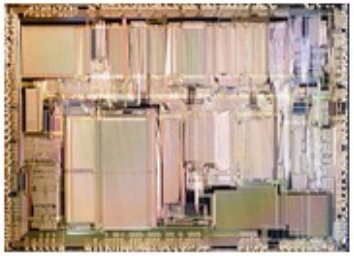
Die photos

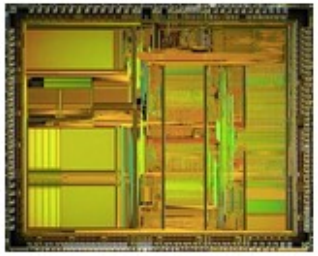

Intel 80960MX


Intel 80960KA


Intel 80960SA


Intel 80960CA


Intel 80960CF


Intel 80960JA

Intel N80960SA (PLCC Package)

Intel GC80960RD66 (BGA Package)

Intel i960HA microprocessor

General Info

Launched	1984
Discontinued	2007[1]
Common manufacturer(s)	Intel

Performance

Max. CPU clock rate	10 MHz to 100 MHz


Intel GC80960RN, sSpec: SL3YW, BGA Package

Intel

i960

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Origin [edit]

Physical sp
Cores
His

The i960 design was begun in response to the failure of Intel's iAPX 432 design of the early 1980s. The iAPX 432 was intended to directly support high-level languages that supported tagged, protected, garbage-collected memory—such as Ada and Lisp—in hardware. Because of its instruction-set complexity, its multi-chip implementation, and design flaws, the iAPX 432 was very slow in comparison to other processors of its time.

In 1984, Intel and Siemens started a joint project, ultimately called BiiN, to create a high-end, fault-tolerant, object-oriented computer system programmed entirely in Ada. Many of the original i432 team members joined this project, although a new lead architect, Glenford Myers, was brought in from IBM. The intended market for the BiiN systems was high-reliability-computer users such as banks, industrial systems, and nuclear power plants.

Architecture [edit]

To avoid the performance issues that plagued the i432, the central i960 instruction-set architecture was a RISC design, which was only implemented in full in the i960MX. The memory subsystem was 33-bits wide—to accommodate a 32-bit word and a "tag" bit to implement memory protection in hardware. In many ways, the i960 followed the original Berkeley RISC design, notably in its use of register windows, an implementation-specific number of caches for the per-subroutine registers that allowed for fast subroutine calls. The competing Stanford University design, MIPS, did not use this system, instead relying on the compiler to generate optimal subroutine call and return code. In common with most 32-bit designs, the i960 has a flat 32-bit memory space, with no memory segmentation, except for the i960MX, which could support up to 2^{26} "objects", each up to 2^{32} bytes in size.[4] The i960 architecture also anticipated a superscalar implementation, with instructions being simultaneously dispatched to more than one unit within the processor.

WIKIPEDIA
The Free Encyclopedia

Intel ARMv5 XScale

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

XScale is a microarchitecture for central processing units initially designed by Intel implementing the ARM architecture (version 5) instruction set. XScale comprises several distinct families: IXP, IXC, IOP, PXA and CE (see more below), with some later models designed as system-on-a-chip (SoC). Intel sold the PXA family to Marvell Technology Group in June 2006.[1] Marvell then extended the brand to include processors with other microarchitectures, like ARM's Cortex.

The XScale architecture is based on the **ARMv5TE** ISA without the floating-point instructions. XScale uses a seven-stage integer and an eight-stage memory super-pipelined microarchitecture. It is the successor to the Intel StrongARM line of microprocessors and microcontrollers, which Intel acquired from DEC's Digital Semiconductor division as part of a settlement of a lawsuit between the two companies. Intel used the **StrongARM** to replace its ailing line of outdated RISC processors, the i860 and i960

All the generations of XScale are 32-bit **ARMv5TE** processors manufactured with a 0.18 µm or 0.13 µm (as in IXP43x parts) proce have a 32 KB data cache and a 32 KB instruction cache.

ISA

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

MIPS

- MIPS I (32-bit) [R2000/3000]
- MIPS32 (32-bit, *MARS*)
- MIPS III (64-bit) [R4000]
- MIPS64 (64-bit)
  - Superset of 32-bit ISA
  - Adds 64-bit ops (“Double”)

MIPS

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wiki

MIPS microprocessors [edit]

The first MIPS microprocessor, the **R2000**, was announced in 1985. It added multiple-cycle multiply and divide instructions in a somewhat independent on-chip unit. New instructions were added to retrieve the results from this unit back to the register file; these result-retrieving instructions were interlocked.

The R2000 could be booted either big-endian or little-endian. It had thirty-one 32-bit general purpose registers, but no condition code register (the designers considered it a potential bottleneck), a feature it shares with the AMD 29000 and the Alpha. Unlike other registers, the program counter is not directly accessible.

The R2000 also had support for up to four co-processors, one of which was built into the main CPU and handled exceptions, traps and memory management, while the other three were left for other uses. One of these could be filled by the optional **R2010** FPU, which had thirty-two 32-bit registers that could be used as sixteen 64-bit registers for double-precision.

MIPS

Jeff Drobman
©2016-23

MIPS Technologies

From Wikipedia, the free encyclopedia
(Redirected from MIPS Computer)

MIPS Technologies, Inc., formerly **MIPS Computer Systems, Inc.**, was an American fabless semiconductor design company that is most widely known for developing the MIPS architecture and a series of RISC CPU chips based on it.[1][2] MIPS provides processor architectures and cores for digital home, networking, embedded, Internet of things and mobile applications.[3][4]

MIPS Technologies, Inc. is owned[5] by Wave Computing, who acquired it from Tallwood MIPS Inc., a company indirectly owned by Tallwood Venture Capital. Tallwood bought it on 2017-10-25 from Imagination Technologies, a UK-based company best known for their PowerVR graphics processor family.[6] Imagination Technologies had previously bought MIPS after CEVA, Inc. pulled out of a bidding on 2013-02-08.

MIPS Technologies, Inc.

MIPS TECHNOLOGIES
Logo

The former MIPS Technologies building in Santa Clara

Type	Subsidiary
Industry	RISC microprocessors
Fate	Acquired in 2018 by Wave Computing
Founded	1984; 36 years ago
Founder	John L. Hennessy
Defunct	2013
Headquarters	Sunnyvale, California, U.S.
Key people	Sandeep Vij
Products	Semiconductor intellectual property
Number of employees	up to 50 (according to LinkedIn in May 2018), previously 146 (September 2010)
Parent	Wave Computing

MIPS

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

History [edit]

MIPS Computer Systems Inc. was founded in 1984[7][8] by a group of researchers from Stanford University that included John L. Hennessy and Chris Rowen. These researchers had worked on a project called MIPS (for *Microprocessor without Interlocked Pipeline Stages*), one of the projects that pioneered the RISC concept. Other principal founders were Skip Stritter, formerly a Motorola technologist, and John Moussouris, formerly of IBM.[9]

The initial CEO was Vaemond Crane, formerly President and CEO of Computer Consoles Inc., who arrived in February 1985 and departed in June 1989. He was replaced by Bob Miller, a former senior IBM and Data General executive. Miller ran the company through its IPO and subsequent sale to Silicon Graphics.

Defunct	2013
Headquarters	Sunnyvale, California, U.S.
Key people	Sandeep Vij
Products	Semiconductor intellectual property
Number of employees	up to 50 (according to LinkedIn in May 2018), previously 146 (September 2010)
Parent	Wave Computing
Website	www.mips.com

In 1988, MIPS Computer Systems designs were noticed by Silicon Graphics (SGI) and the company adopted the MIPS architecture for its computers.[10] A year later, in December 1989, MIPS held its first IPO. That year, Digital Equipment Corporation (DEC) released a Unix workstation based on the MIPS design.

After developing the R2000 and R3000 microprocessors, a management change brought along the larger dreams of being a computer vendor. The company found itself unable to compete in the computer market against much larger companies and was struggling to support the costs of developing both the chips and the systems (MIPS Magnum). To secure the supply of future generations of MIPS microprocessors (the 64-bit R4000), SGI acquired the company in 1992[11] for $333 million[12][13] and renamed it as MIPS Technologies Inc., a wholly owned subsidiary of SGI.[14]

During SGI's ownership of MIPS, the company introduced the R8000 in 1994 and the R10000[15] in 1996 and a follow up the R12000 in 1997.[16] During this time, two future microprocessors code-named *The Beast* and *Capitan* were in development; these were cancelled after SGI decided to migrate to the Itanium architecture[17] in 1998.[12][18] As a result, MIPS was spun out as an intellectual property licensing company, offering licences to the MIPS architecture as well as microprocessor core designs.

MIPS

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Company timeline [edit]

Year	
1981	Dr. John Hennessy at Stanford University founds and leads Stanford MIPS, a research program aimed at building a microprocessor using RISC principles.
1984	MIPS Computer Systems, Inc. co-founded by Dr. John Hennessy, Skip Stritter, and Dr. John Moussouris[43]
1986	First product ships: R2000 microprocessor, Unix workstation, and optimizing compilers
1988	R3000 microprocessor
1989	First IPO in November as MIPS Computer Systems with Bob Miller as CEO
1991	R4000 microprocessor
1992	SGI acquires MIPS Computer Systems. Transforms it into internal MIPS Group, and then incorporates and renames it to MIPS Technologies, Inc. (a wholly owned subsidiary of SGI)
1994	R8000 microprocessor
1994	Sony PlayStation released, using an R3000 CPU with custom GTE coprocessor
1996	R10000 microprocessor; Nintendo 64 released, incorporating a cut down R4300 processor.
1998	Re-IPO as MIPS Technologies, Inc
1999	Sony PlayStation 2 released, using an R5900 cpu with custom vector coprocessors
2002	Acquires Algorithmics Ltd, a UK-based MIPS development hardware/software and consultancy company.
September 6, 2005	Acquires First Silicon Solutions (FS2), a Lake Oswego, Oregon company as a wholly owned subsidiary. FS2 specializes in silicon IP, design services and OCI (On-Chip Instrumentation) development tools for programming, testing, debug and trace of embedded systems in SoC, SOPC, FPGA, ASSP and ASIC devices.
2007	MIPS Technologies acquires Portugal-based mixed-signal intellectual property company Chipidea
February 2009	MIPS Joins Linux Foundation[44]
May 8, 2009	Chipidea is sold to Synopsys.
June 2009	Android is ported to MIPS[45]

MIPS

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wiki

R3000: 32-bit CPU → *pipelined* (5 stages)

The **R3000** succeeded the R2000 in 1988, adding 32 KB (soon increased to 64 KB) caches for instructions and data, along with support for shared-memory multiprocessing in the form of a cache coherence protocol. While there were flaws in the R3000s multiprocessing support, it was successfully used in several successful multiprocessor computers. The R3000 also included a built-in MMU, a common feature on CPUs of the era. The R3000, like the R2000, could be paired with a **R3010** FPU. The R3000 was the first successful MIPS design in the marketplace, and eventually over one million were made. A speed-bumped version of the R3000 running up to 40 MHz, the **R3000A** delivered a performance of 32 VUPs (VAX Unit of Performance). The MIPS R3000A-compatible **R3051** running at 33.8688 MHz was the processor used in the Sony PlayStation though it didn't have FPU or MMU. Third-party designs include Performance Semiconductor's **R3400** and IDT's **R3500**, both of them were R3000As with an integrated R3010 FPU. Toshiba's **R3900** was a virtually first SoC for the early handheld PCs that ran Windows CE. A radiation-hardened variant for space applications, the Mongoose-V, is a R3000 with an integrated R3010 FPU.

The **R4000** series, released in 1991, extended MIPS to a full 64-bit architecture, moved the FPU onto the main die to create a single-chip microprocessor, and had a high clock frequency of 100 MHz at introduction. However, in order to achieve the clock frequency, the caches were reduced to 8 KB each and they took three cycles to access. The high operating frequencies were achieved through the technique of deep pipelining (called super-pipelining at the time). The improved **R4400** followed in 1993. It had larger 16 KB primary caches, largely bug-free 64-bit operation, and support for a larger L2 cache.

MIPS, now a division of SGI called MTI, designed the low-cost **R4200**, the basis for the even cheaper **R4300i**. A derivative of this microprocessor, the NEC VR4300, was used in the Nintendo 64 game console.[1]

R4000: 1st 64-bit CPU → *super-pipelined* (8 stages)

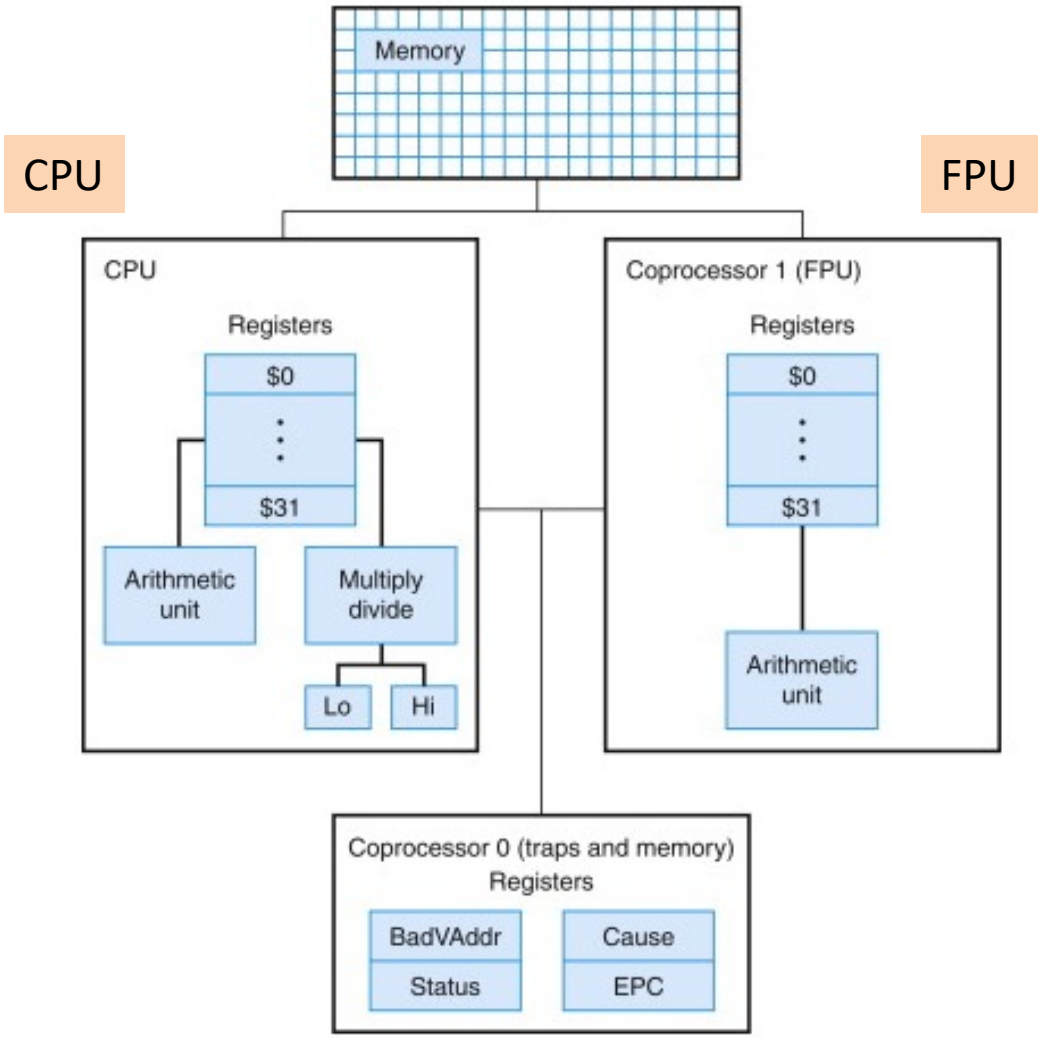
Quantum Effect Devices (QED), a separate company started by former MIPS employees, designed the **R4600** *Orion*, the **R4700** *Orion*, the **R4650** and the **R5000**. Where the R4000 had pushed clock frequency and sacrificed cache capacity, the QED designs emphasized large caches which could be accessed in just two cycles and efficient use of silicon area.

MIPS I– Base (R2000) Org

Jeff Drobman
©2016-23

Hennessy & Patterson MIPS

Figure 7.10.1: MIPS R2000 CPU and FPU (COD Figure A.10.1).



IDT's MIPS R3000 Die

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

First MIPS RISC CPUs

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

32-bit

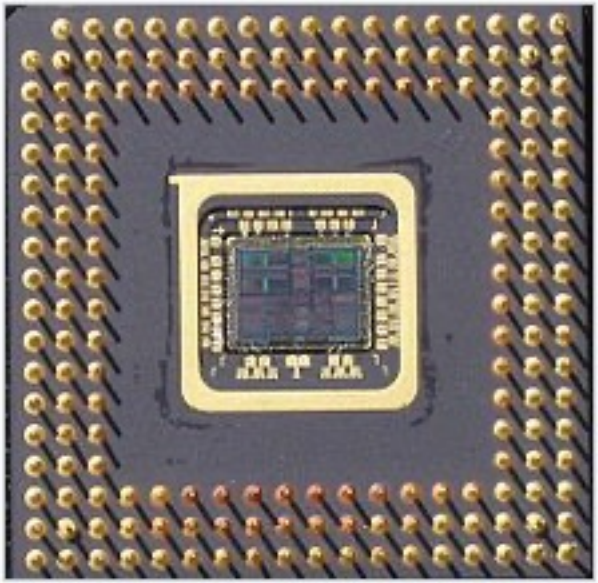
64-bit

MIPS

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wiki

R4700



Bottom-side view of package of R4700 Orion with the exposed silicon chip, fabricated by IDT, designed by Quantum Effect Devices

Top-side view of package for R4700 Orion

Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Advanced RISC Microprocessors

- Apple A14/M1
- Intel Core i3/5/7/9
- AMD Zen 3
- Mobile SoC's

Apple M1

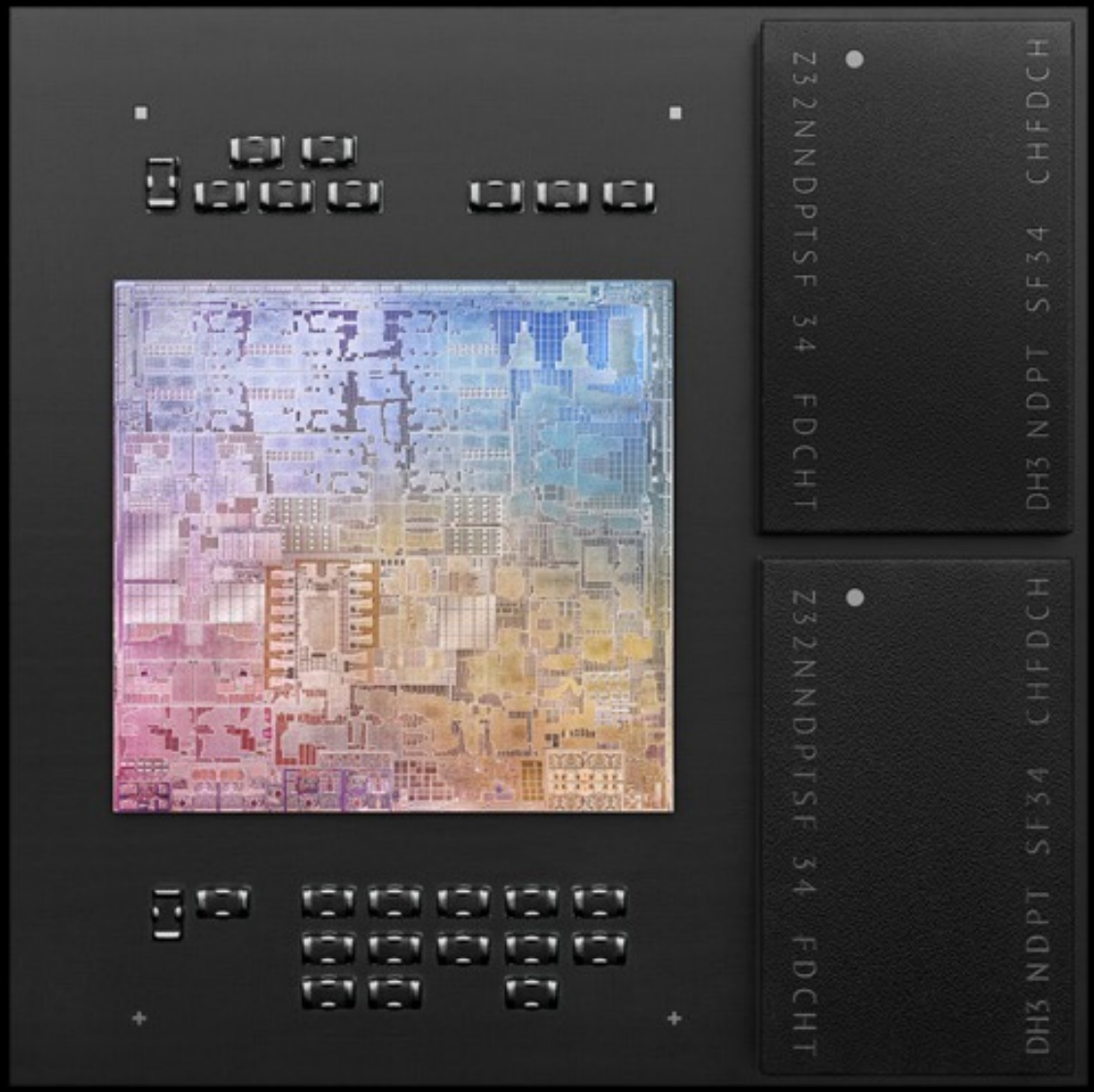
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

5-nanometer process

The first personal computer chip built with this cutting-edge technology.

16 billion transistors

The most we've ever put into a single chip.



Apple Event

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

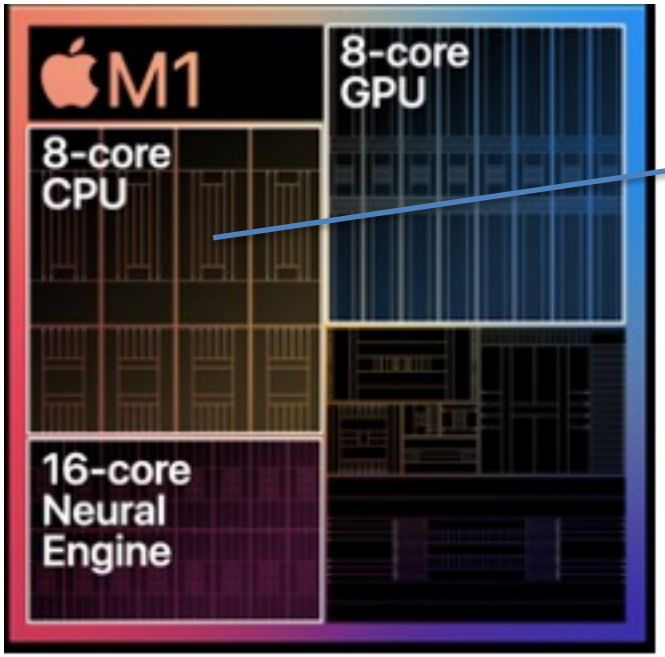
November 10, 2020

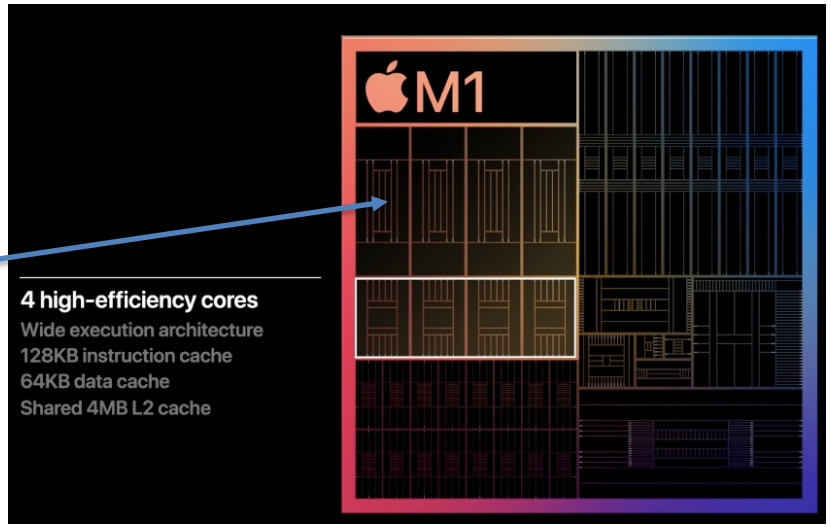
11 trillion
Operations per second

11 Tera FLOPS

- Cores
  - 8 CPU
  - 8 GPU
  - 16 NPU

- CPU cores
  - 4 Hi Perf (20W)
  - 4 Hi Efficiency (1.3W low power)





AMD vs Intel: CPU Families

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Market Segment	AMD	Intel
Desktop	Ryzen 4K/Athlon 3K	Core i7/i9 (10th gen)
Laptop	Ryzen 4000	Ice Lake
Gaming	Ryzen Threadripper +Radeon	Core Extreme
Server/Workstn	Epyc	Xeon

According to the company, the AMD Ryzen 4700 G series desktop processor offers up to 2.5x multi-threaded performance compared to the previous generation, up to 5% greater single-thread performance than the Intel Core i7-9700, up to 31% greater multithreaded performance than the Intel Core i7-9700, and up to 202% better graphics performance than the Intel Core i7-9700.

Intel New Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Why doesn't Intel have as strong of integrated graphics on their CPUs, such as their Intel UHD 630 graphics, compared to AMD's Vega 11 integrated graphics?

Brett Bergan, Building PC's for 25 years
Answered 48m ago

Unfortunately for AMD fans, Vega 11 was a great product that found its way to ONE processor (actually two if you consider the 2400G and 3400G two different CPUs)

But that detail aside, Intel has been recycling the same 14nm "Skylake" HD 530/630 GPU for five years already. I have a sneaky suspicion that 10th gen Comet Lake CPU models consist of a lot of recycled Coffee Lake silicon that didn't get sold in 2018. The i3–10100 hyperthreaded quad is essentially a i7–7700 that has a locked multiplier set at 4.3GHz

Same CPU. Same GPU. Just three generations later.

10nm Ice Lake with its somewhat improve ... (more)

AMD Ryzen

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Norman Latifov, knows Turkish
Answered 2h ago

It is the latest mobile CPU from AMD. Ryzen 4000 CPUs are only available on laptops and they are the fastest mobile CPUs available right now. I am using a Lenovo Yoga slim 7 with r7 4800u and before I bought it I did a lot of research. Based on reviews, they are even faster than 11th gen Intel CPUs that are yet to come. Although their integrated GPU is not as good as iris graphics (Intel 11th gen CPUs' integrated GPU), vega series GPUs are still a good option. And in my opinion, ryzen 4000 CPUs have a great multicore performance.

Zachary Hawkshaw, AMD Hardware Connoisseur
Answered 10h ago

I don't know where you got your information, but that's not true. The 2700X only has 4.8 billion transistors while the 3700X has 19.2 billion.

Transistor count doesn't necessarily mean more performance by itself. The 3700X is better because it has a newer architecture (Zen2 vs Zen+) with improvements to the Infinity Fabric, as well as a higher turbo frequency.

AMD Ryzen

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

John B. Anderson III, IT Consultant, PC Integrator, 20+ Years in IT and Gaming.

Answered Mon

Well, it's only on the Laptop side... Since they skipped it to make them both match desktop/laptop for Zen 3 architecture.

Laptops with the 4XXX name are actually Zen 2 processors.

See some guy in marketing thought it'd be a good idea to call Zen processors on laptops 2XXX series, and so when the 2XXX series came out on desktop the laptop was already at 2XXX so they called them 3XXX.

Example Desktop CPU Ryzen 5 2600x the laptop version would be a Ryzen 5 3550h

AMD Ryzen 5 2600 vs 3550H with the typical lower performance on laptop vs desktop CPU. Same Zen+ architecture in both.

On the Zen 2 Archetuxture

AMD Ryzen 5 3600 vs 4600HS we see the Ryzen 5 4600 HS ~5% of the speed.

There are no Zen 3 laptop processors as of the time I'm writing 11/2020

Typically we call it a "Gen / Generation" when they name a model with a change in the first digit.

Example Core i5 10XXX would be a 10th gen i5. Ryzen 7 1800x would be a 1st Gen Ryzen, However, Since the laptops, 4XXX were the only "Gen 4s" but they were technically Gen 3's well the answer is somewhat tricky.

AMD vs Intel

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Sockets

The current Threadripper sTRX4 socket is an LGA design with 4094 pins.. they're kind of mindboggling to look at. Modern EPYC processors use a mechanically identical but electrically tweaked Socket SP3r3 socket. Older chips use TR4 and SP3r2 sockets, respectively. The EPYC series and Threadripper Pro actually support up to 2TiB DDR4 DRAM on eight channels and 128 PCI Express 4.0 links. Today's standard Threadrippers support 128 GiB or 256GiB DDR4 DRAM on four channels, with up to 88 PCI Express 4.0 links, and of course, up to 64 CPU cores on all three platforms.

So you're looking for an Intel processor for consumers/workstations that's more or less similar to Threadripper. Intel's answer to EPYC is Xeon, so are there any mainstream i-series processors that correspond closely? The latest high-end

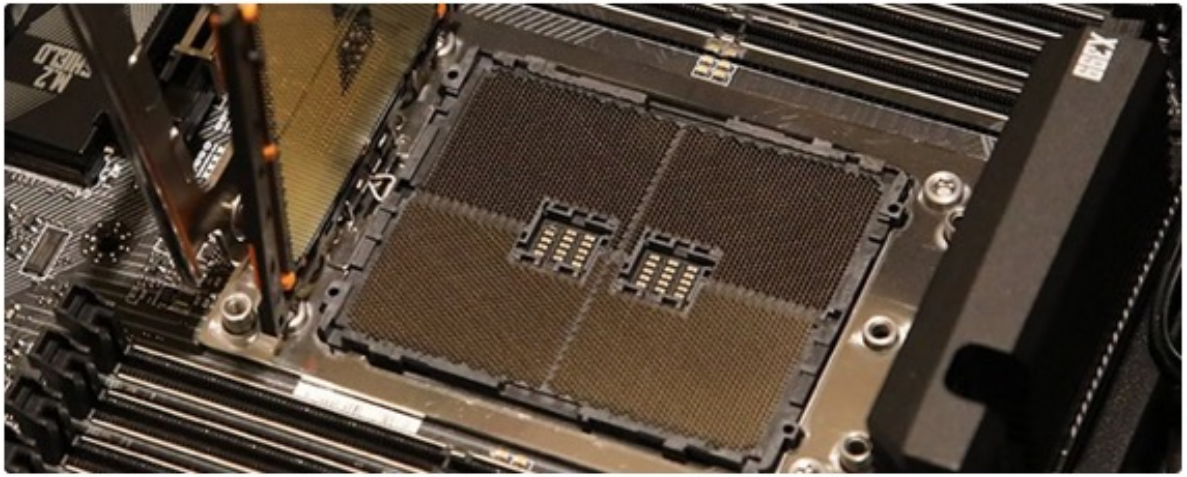
AMD vs Intel

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Sockets

AMD Threadrippers are essentially consumer versions of the EPYC line of server processors. There are differences, but the basic idea is the same: four DDR4 memory channels, high core count, etc. My aging Threadripper system has "only" sixteen processor cores and the usual four 64-bit DDR4 memory channels.

Like all Ryzen family processors, Threadrippers are made of multiple "chiplets" connected by ultra high speed Infinity Fabric links. Each chiplet so far contains up to eight processor cores. The central chip in generation 2 and later Threadrippers is a I/O chip, supporting PCI Express links, that sort of thing.



The current Threadripper sTRX4 socket is an LGA design with 4094 pins.. they're kind of mindboggling to look at. Modern EPYC processors use a mechanically identical but electrically tweaked Socket SP3r3 socket. Older chips use TR4 and SP3r2 sockets, respectively. The EPYC series and Threadripper Pro actually support up to 2TiB DDR4 DRAM on eight channels and 128 PCI Express 4.0 links. Today's standard Threadrippers support 128 GiB or 256GiB DDR4 DRAM on four channels, with up to 88 PCI Express 4.0 links, and of course, up to 64 CPU cores

AMD vs Intel i9

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

So you're looking for an Intel processor for consumers/workstations that's more or less similar to Threadripper. Intel's answer to EPYC is Xeon, so are there any mainstream i-series processors that correspond closely? The latest high-end Xeons and Phi processors use Intel's LGA 3647 socket. This socket supports six channels of DDR4 memory, but there is no consumer version of an LGA3647 processor.

So the Intel answer to complete with Threadripper is the LGA2066 socket, also called Socket R4. There are lower-end Xeons that also use this socket. This supports DDR4 up to 256GiB on four channels, 48 PCI Express 3.0 lanes (with an additional 24 PCI Express 3.0 links in the X299 chipset). Current LGA2066 chips offer up to 18 CPU cores.

RISC-V

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Heikki Kultala, Technical leader, SoC architecture at Nokia (2020-present)

Answered February 6

No, RISC-V is 1980s done correctly, 30 years later.

It still concentrates on fixing those problems that we had in 1980s (making instruction set that is easy to pipeline with a simple pipeline), but we mostly don't have anymore, because we have managed to find other, more practical solutions to those problems.

And it's "done correctly" because it abandons the most stupid RISC features such as delay slots. But it ignores most of the things we have learned after that.

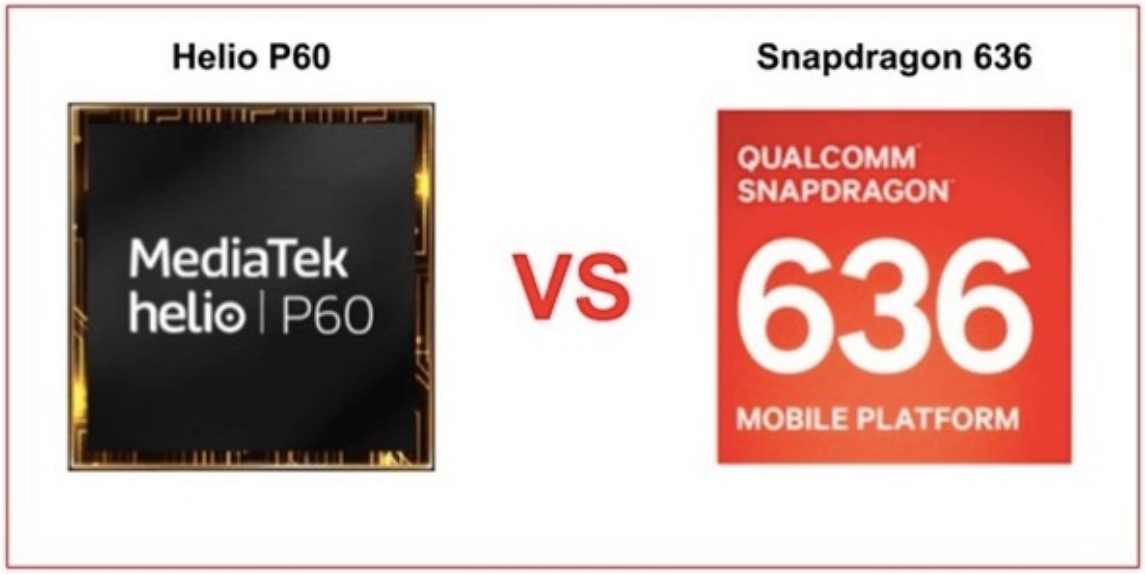
ARMv8 is much more advanced and better instruction set which makes much more sense from a technical point of view. Many common things require much more RISC-V instruction than ARMv8 instructions. The only good reason to use RISC-V instead of ARM is to avoid paying licence fees to ARM.

MediaTek vs. Snapdragon

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

MediaTek Helio P60

Built on the 12nm fabrication process, MediaTek Helio P60 is the upper-mainstream processor of the MediaTek introduced in 2008 mainly for android. The processor is equipped with 4x big ARM Cortex-A73 cores and 4x small and power-efficient ARM Cortex-A53 cores in two clusters. The cores' clusters have the ability to clock the speed up to 2 GHz. The processor also integrates an ARM Mali-G72MP3 GPU and a dedicated AI processing unit.



Snapdragon 636

Built on 14nm Fabrication process, Snapdragon 636 was launched at the same time with eight cores based on Kryo 260 cores ticking at up to 1.8 GHz. It used Adreno 590 as the GPU. The cores of the processor are customizable and hence needed to be

X-Large Dice

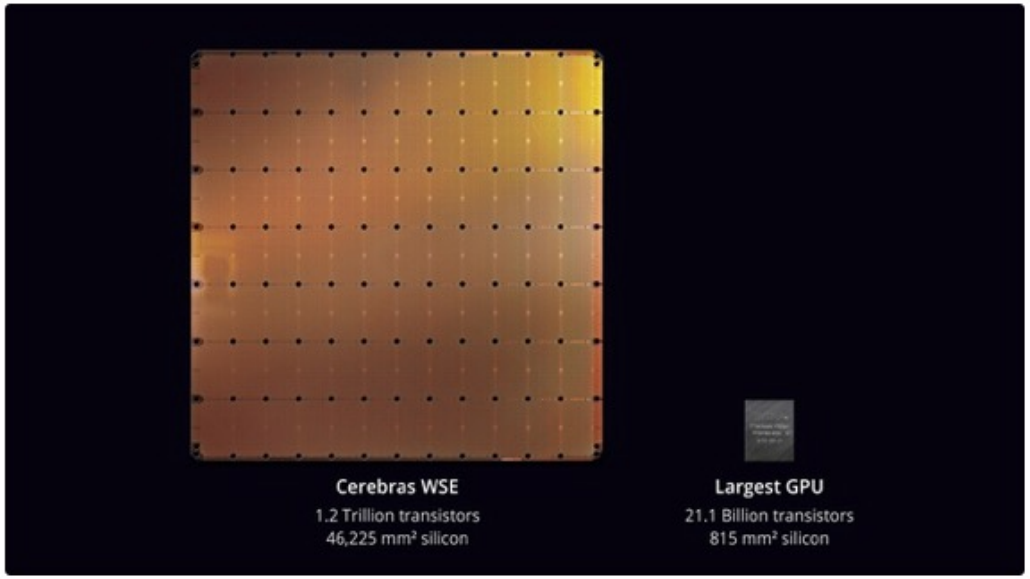
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Vladislav Zorov · 10h ago

RTX 3080 and 3090 have 28.3 billion transistors, on a 628 mm² die.

RX 6800 XT has 26.8 billion transistors, on a 520 mm² die.

In any case, M1 is definitely not the leader. The leader is this thing, at 1.2 *trillion* transistors:



(note that even the old GPU they're comparing it to in the picture had 21 bn transistors - and the new ones have more - so no way 16 bn is the leader)

wasn't aware of the 21B transistor chip, but it is way larger than a 600 sq mm die (about 1 inch square). that large of a die is likely to be way expensive due to defect densities and silicon wafer costs. Apple still leads in density by using 5nm instead of AMD using 7nm. and that monster "chip" is 8.5 inches square — more a board size than a "chip".

Reply

1 sq in
645 sq mm

1 inch
25.4 mm

Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Moore's Law

(see separate slide set ***Chips & Fabs***)

Gordon Moore

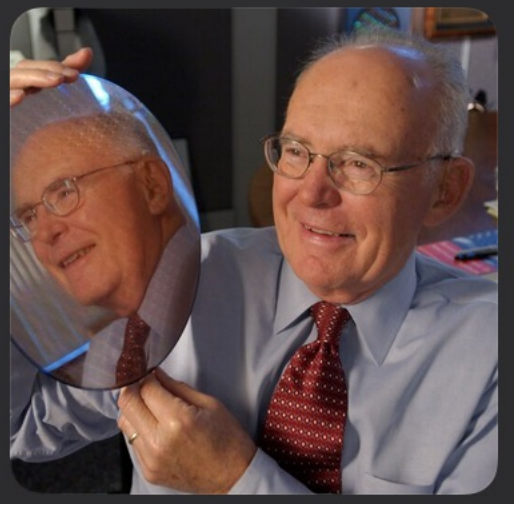
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Jan 3, 1929 -- March 24, 2023

News+

Los Angeles Times

Gordon E. Moore, Intel founder and creator of Moore's Law, dies at 94



Gordon Moore

Moore in 1978

Born	Gordon Earle Moore January 3, 1929 Pescadero, California, U.S.[1]
Died	March 24, 2023 (aged 94) Waimea, Hawaii, U.S.
Education	University of California, Berkeley (BS) California Institute of Technology (PhD)

Chips-Moore's Law

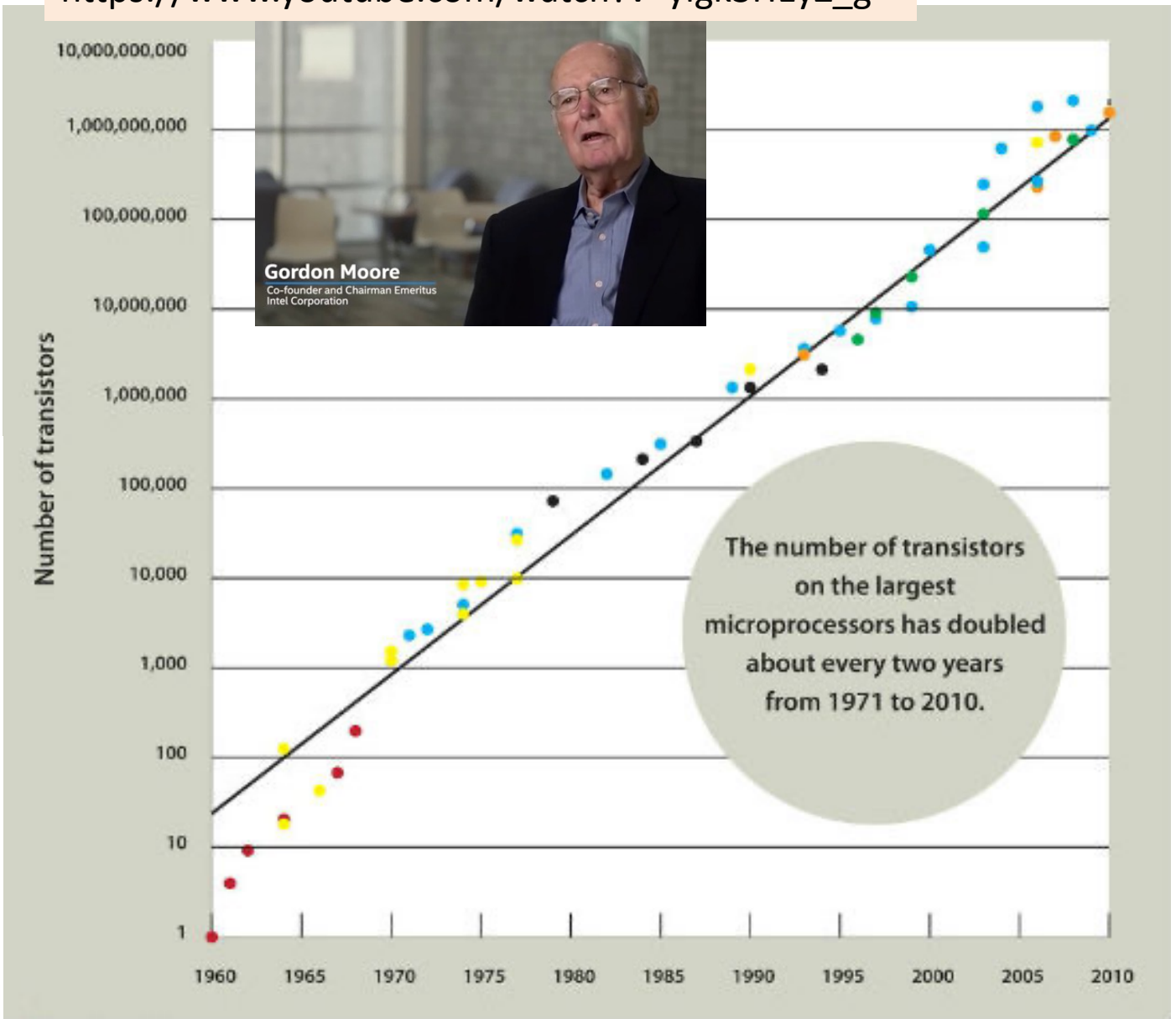
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

https://www.youtube.com/watch?v=ylgk3HEyZ_g

transistors doubles every 2 years

Plaque to Moore\'s Law at the technology plaza in Mountain View, beneath a model of the Silicon crystal

Dave Laws (2018)

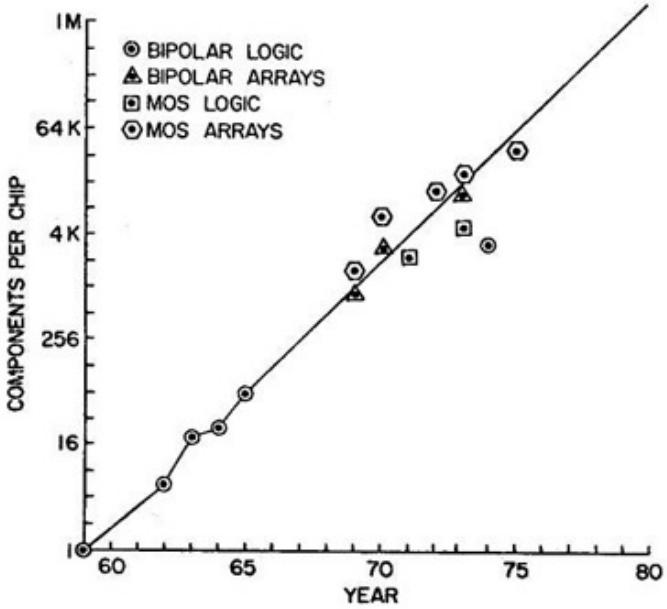


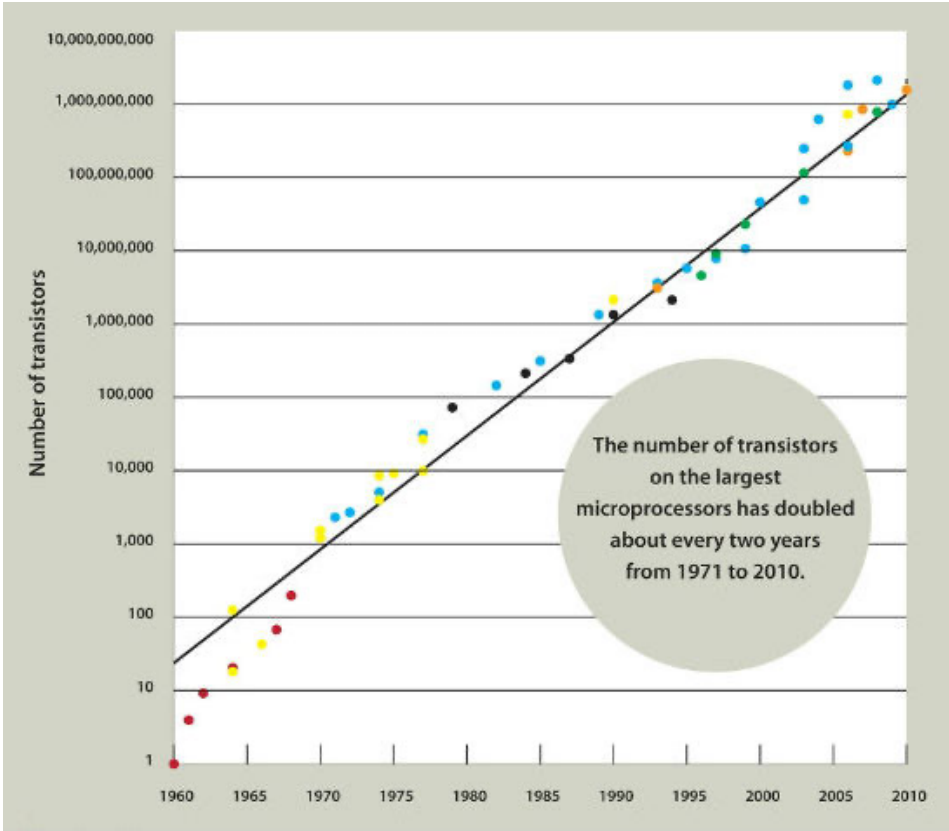
Moore's Law

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Looking Back

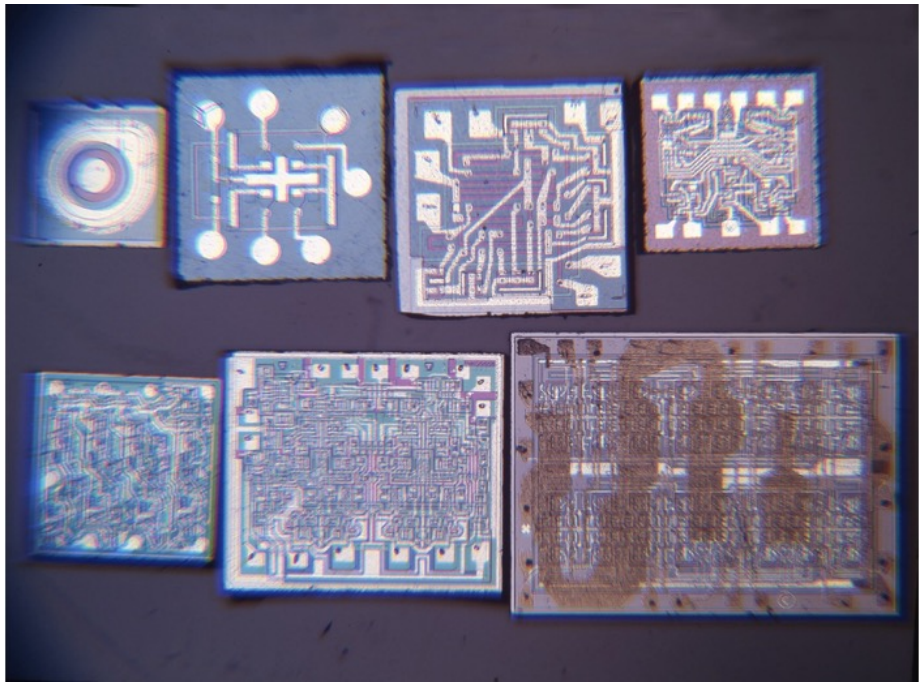
- Original in 1965: # Transistors will double **every year** (12 months)
- Moore revised his prediction in 1975: double **every 2 years** (24 months)
  → THIS IS MOORE'S LAW
- Intel's exec David House added CPU complexity would double **every 18 months**
- History shows # Transistors has doubled every –
  - **2 years** in *logic*
  - **18 months** in DRAM/SRAM





Origin of Moore's Law

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



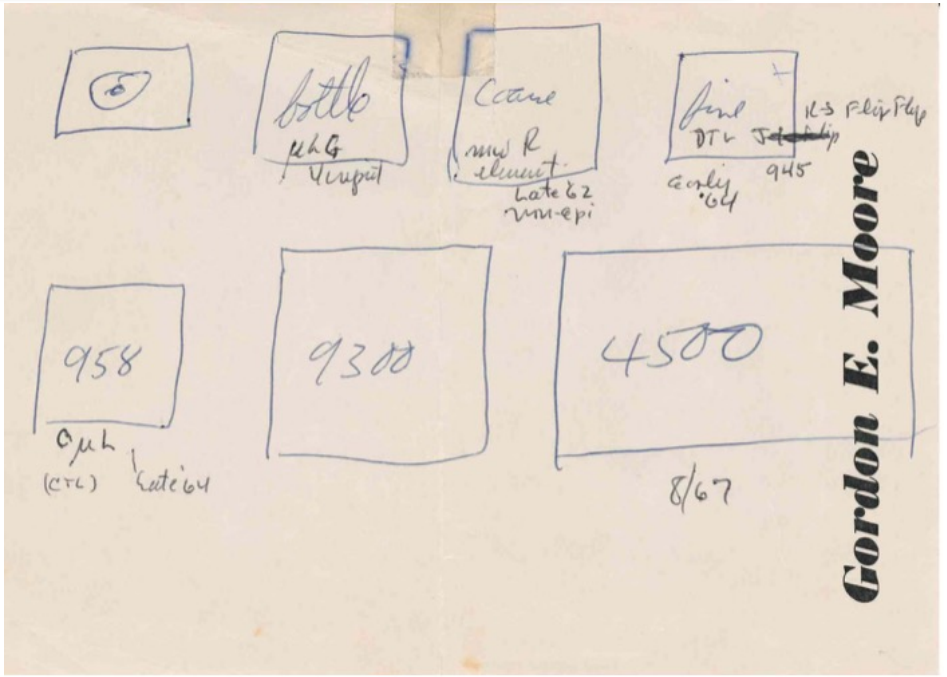


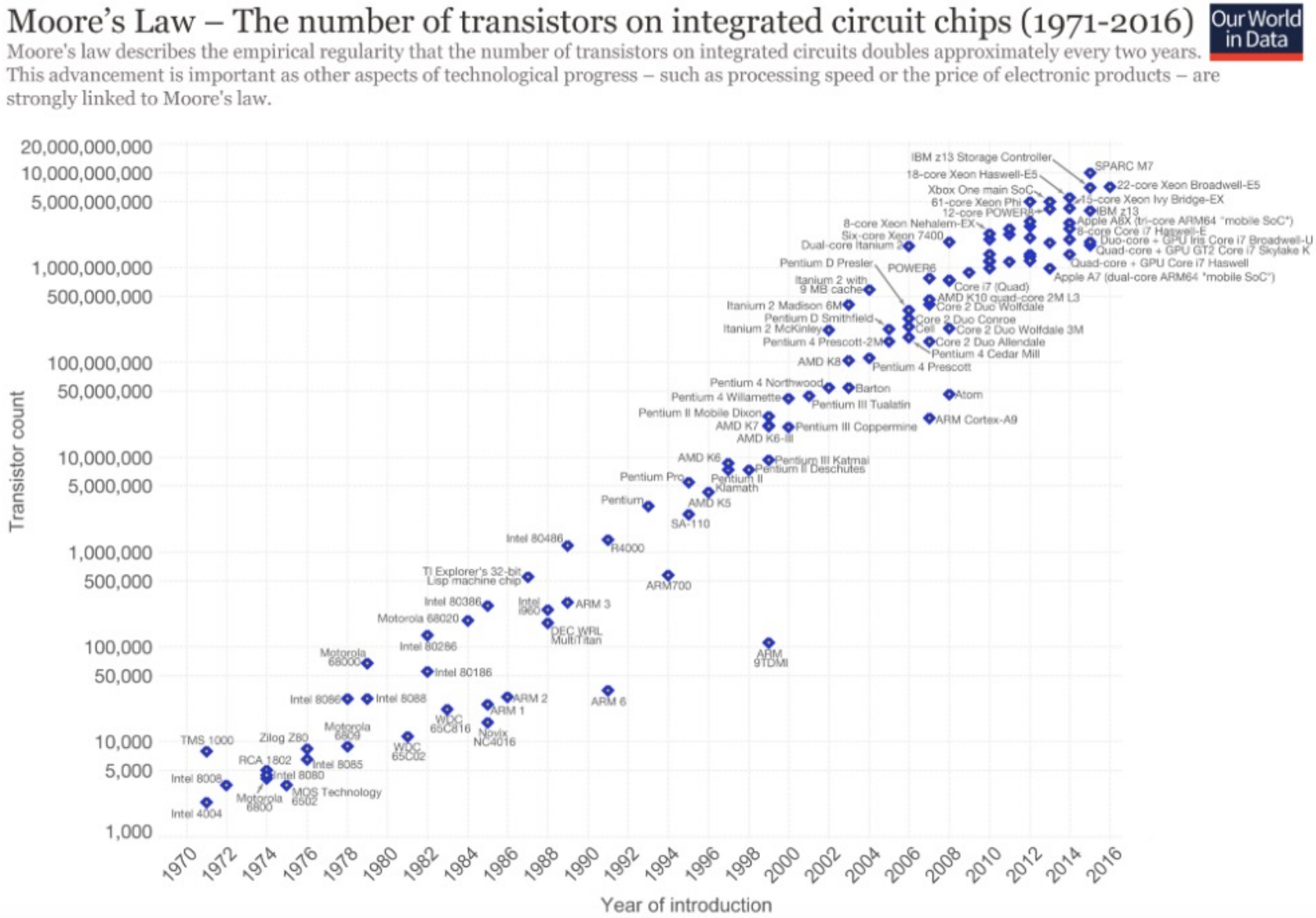
Figure 3. Gordon Moore notes on IC device types. Collection of the Computer History Museum, 102783359.

Year	Device	Function	Transistors	Resistors	Components	LOG_2
1959	2N697	Transistor	1	0	1	0
1962	Type G	RTL 3 - I/P gate	3	4	7	2.8
1963 (late 62)	Type R	RTL D Flip Flop	15	18	33	5.0
1964	945	DTL R-S Flip Flop	13	21	34	5.1
1965 (late 64)	958	RTL Counter	33	25	58	5.9
1966	9300	TTL Shift Register	85	40	125	7.0
1967	4500	DTL 32- Gate Array	200	64	264*	8.0

Figure 4. Table of component count for devices in photograph.

Microprocessor Timeline

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

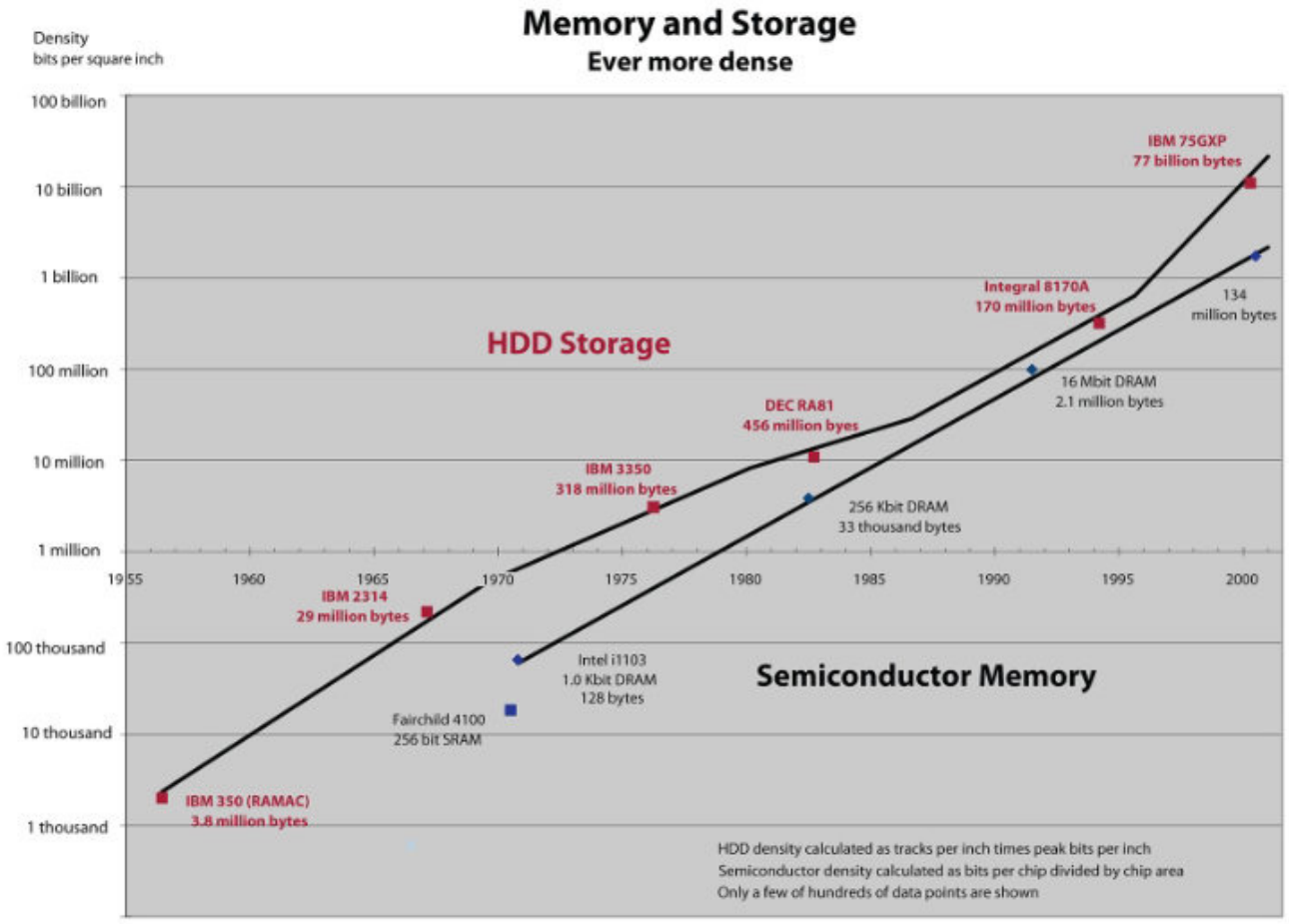


Data source: Wikipedia (https://en.wikipedia.org/wiki/Transistor_count)
The data visualization is available at OurWorldinData.org. There you find more visualizations and research on this topic.

Licensed under CC-BY-SA by the author Max Roser.

Memory Timeline

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



Hard disk storage has become denser at an exponential rate over the last 50 years, just like main memory. The dramatic increase in capacity and speed of both has fueled the increasing power of computers.

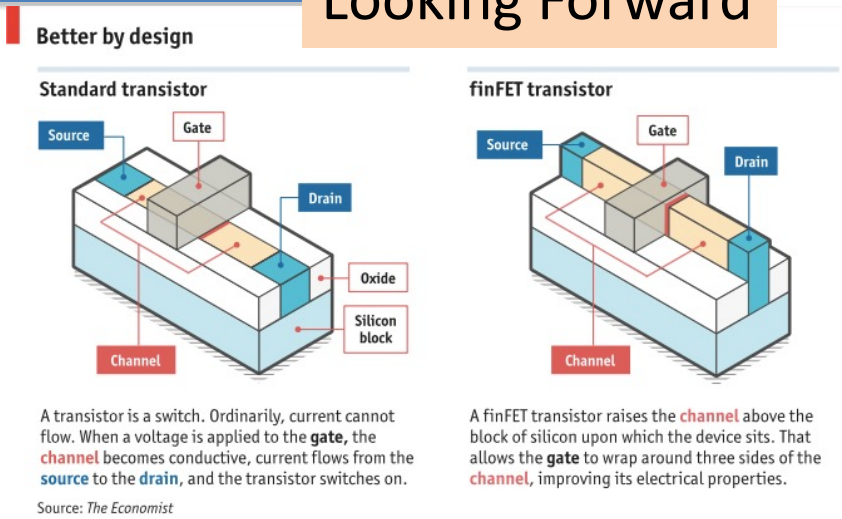
Future of Moore's Law

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

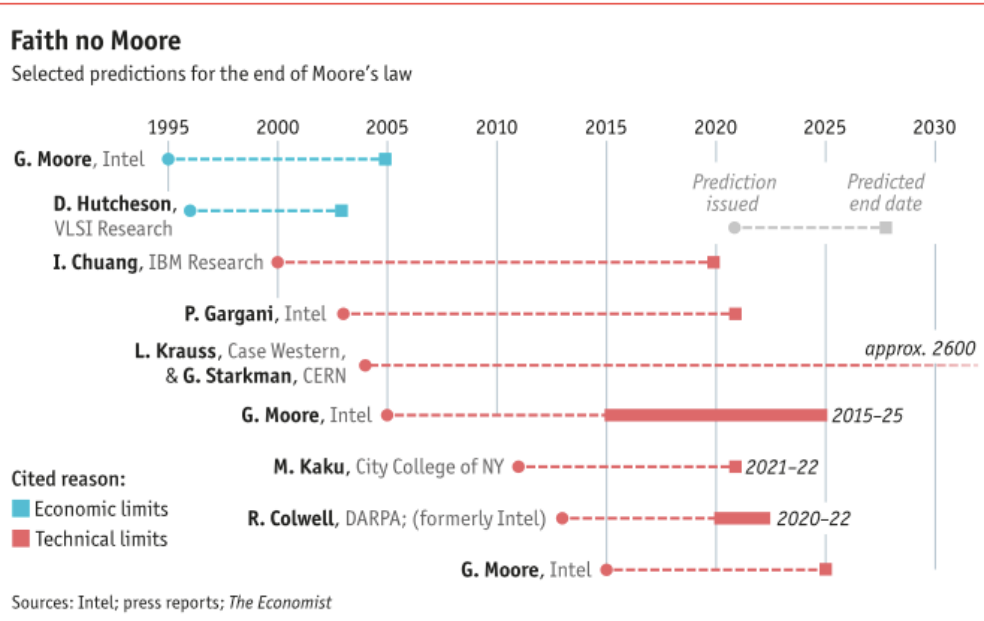
Looking Forward

- Chip design
  - Transistors
    - SiGe
    - FinFET
    - JNT
  - Chip stacks (3D hybrid)
    - Intel/Micron 3D Xpoint
  - 3D
    - NAND Flash (EEPROM)
- Architecture
  - Specialized hardware (GPU, APU, etc.)
  - Reconfigurable hardware (FPGA)
- Thermal/Cooling
  - Microfluidics (liquid cooling)
- Something completely diffferent
  - Molecular computing
  - Quantum computing

"As Moore's Law slows, we are being forced to make tough choices between Power, Performance and Cost." (ARM)



New sorts of transistors can eke out a few more iterations of Moore's law, but they will get increasingly expensive



Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wafer Fabs

(see separate slide set ***Chips & Fabs***)

Wafer Fabs Today

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

1968	❖ Intel
1978	❖ Micron**
1980	❖ Samsung
1987	❖ TSMC* (1st foundry)
2009	❖ AMD → Global Foundries*
2010	❖ Chartered → Global Foundries*
2014	❖ IBM → Global Foundries*
	❖ SMIC* (China)

*Pure Foundry

**Internal use only

Fabs – AMD, Intel

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

AMD Sunnyvale Fab 1 1970

$1M

Cost x10,000 in 40 years
averages to
250x per year

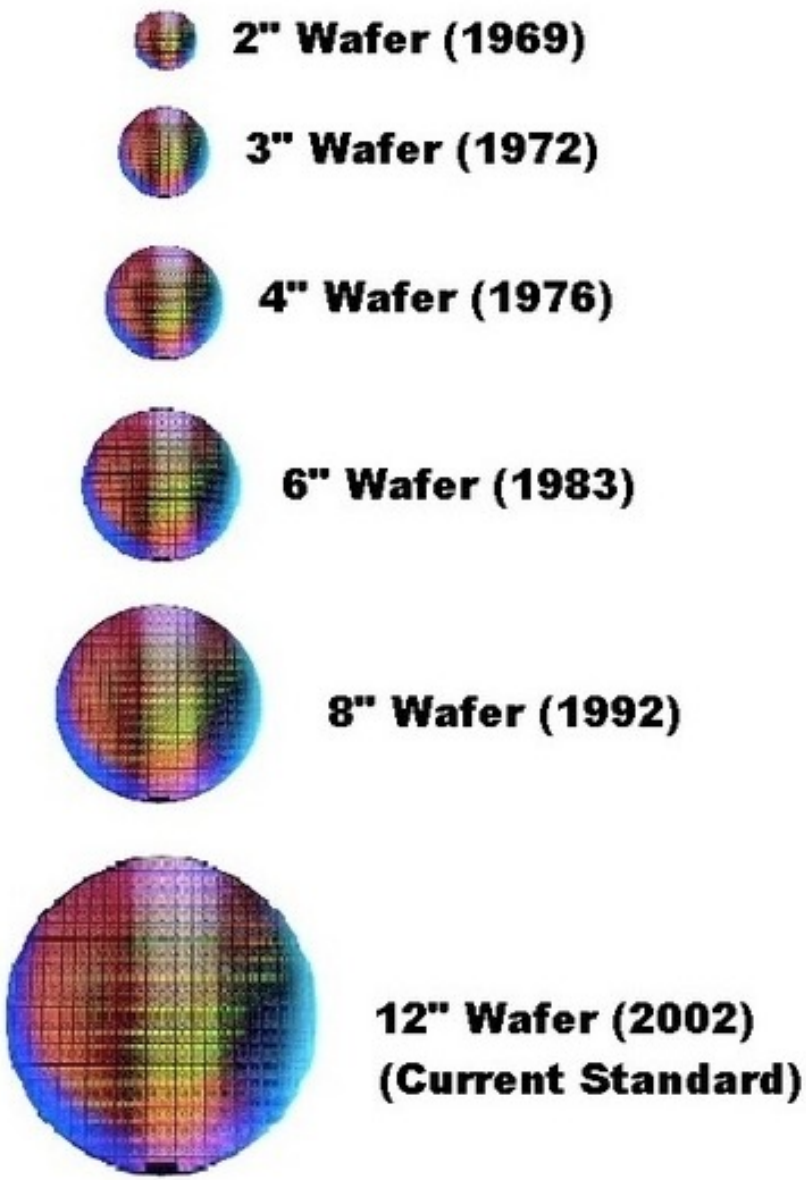
$10B

Intel's latest Fab in Hillsboro

An aerial view of Ronler Acres, Intel's largest silicon research and development hub.

Wafers

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

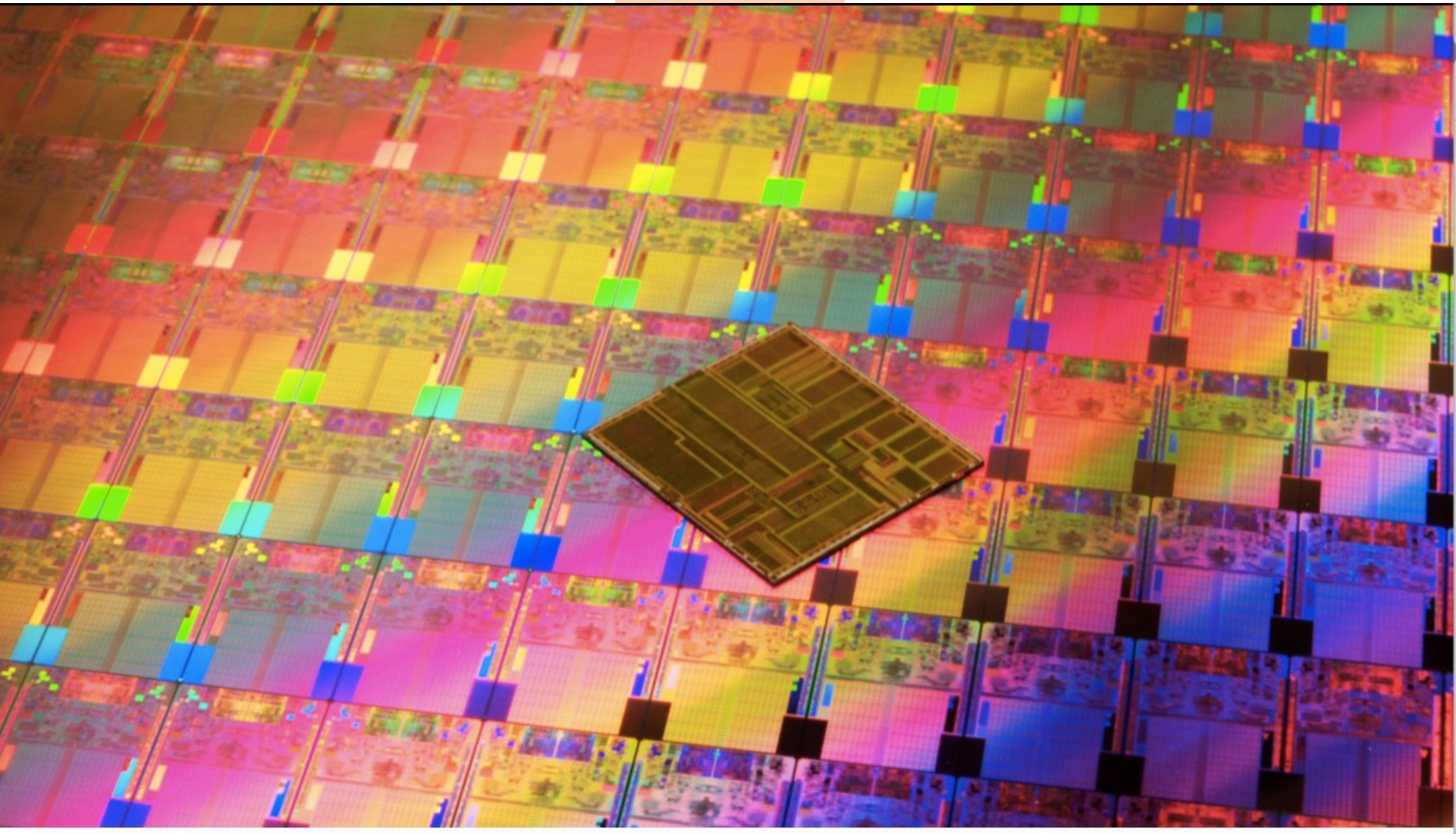


300mm

Die on Wafer

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Intel Pentium

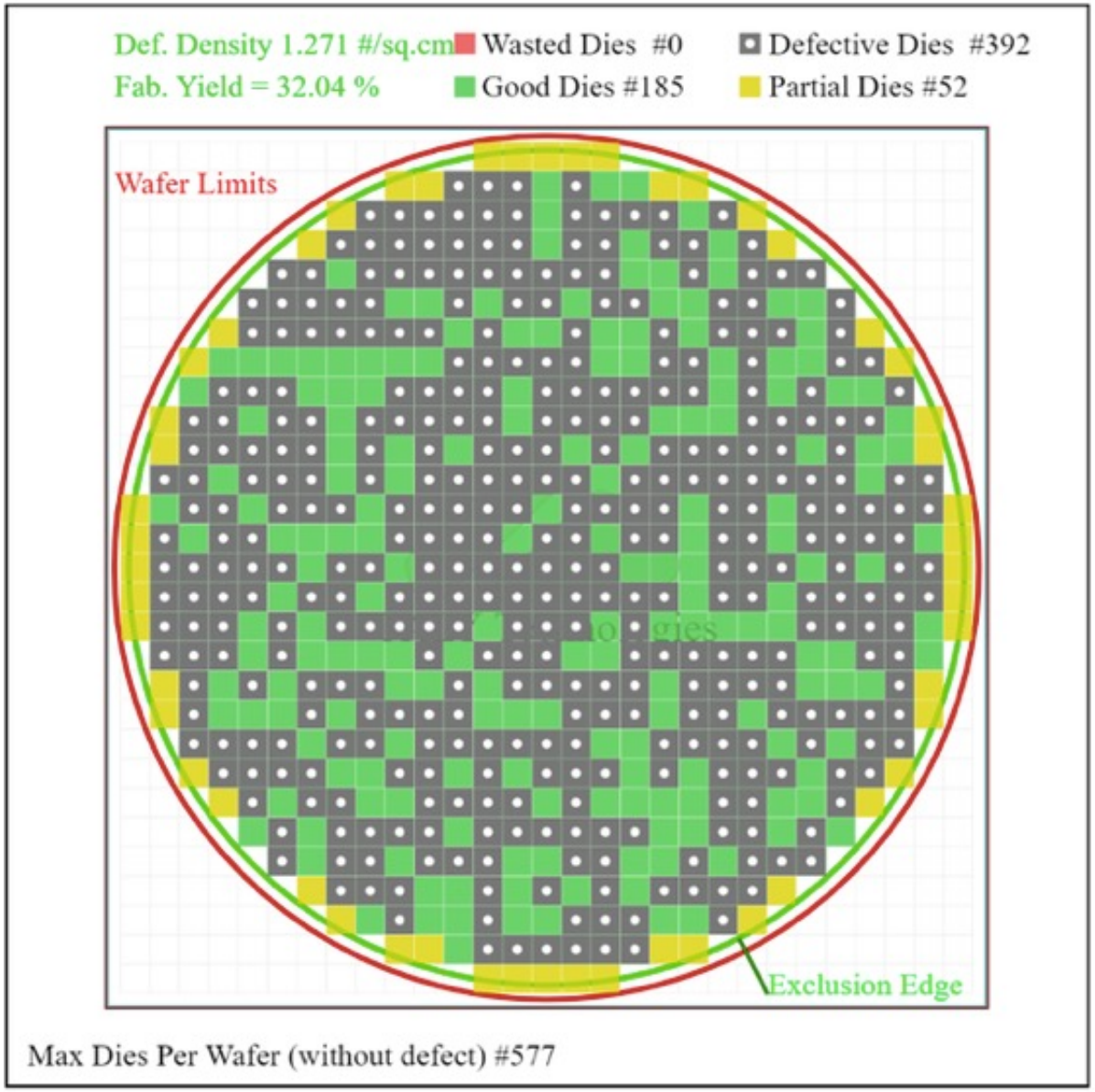


Intel Pentium microprocessor die and wafer

Quora

Wafers

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



Wafers

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Quora

After taking into consideration the wasted dies intersecting the "Exclusion Edge" there are a possible 577 dies produced. In this sample, 392 have defects of some sort, leaving only 185 that can be used as intended.

Assuming these are CPU dies with integrated GPU, any defect in a GPU region can result in a CPU with the GPU disabled and be sold at a lower price. If this is a die with six CPU cores, one pair of cores can be disabled to make it a quad core. In that way, maybe half of the defective chips can be salvaged.

With say 14nm production there are fewer transistors per die, and thus a lower probability of defects as compared to 7nm production. The defect rate goes up because there simply are a lot more transistors and a lot more opportunities for a defect to occur.

The problem is greatly compounded when the die is quite large, because then the possible number of good dies per wafer goes down and the probability of getting defects goes up. GPU dies like those made by Nvidia are easier to bin because multiple compute units can be disabled and still result in a perfectly functional product.

Just as a case in point. A RTX 2080 has 46 compute units, while a RTX 2060 KO has only 30. But they both use the same TU104 silicon. The 2060 KO edition has a whopping 16 of its compute units disabled. Fully 1024 of its GPU cores are dead. You simply can't do that with CPU silicon.

Intel Process Nodes

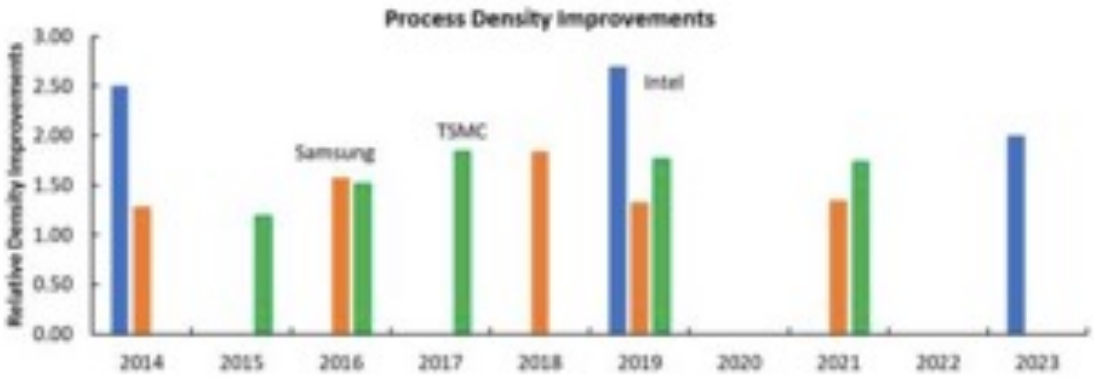
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Slower Node Transitions Versus Foundries

IC KNOWLEDGE LLC

	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023
Intel	14nm					10nm				7nm
Samsung	14nm		10nm		7nm	5nm			3nm	
TSMC		16nm	10nm	7nm		5nm		3nm		2nm?

- Intel takes bigger density jumps but less often.
- TSMC and Samsung take smaller jumps more frequently, 5 nodes versus Intel's 3.



3/24/2021 Copyright IC Knowledge LLC, all rights reserved 4

Figure 4. Node Introductions.

Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Memories

Memory IC Timeline

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

RAM

- ROM– 1st Semiconductor
- DIP packages

1966
- RAM-- Bipolar RAMs (SRAM) introduced
- DRAM– IBM conceives DRAM cell (1T, 1C)

1968
- CMOS SRAM– 1st parts by RCA

1971
- Microprocessor & RAM in MOS invented by Intel

1987
- Toshiba intro's Flash EEPROM,

Memory Types

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wiki

Computer memory types

Volatile

RAM

DRAM (SDRAM · DDR · GDDR · HBM) · SRAM

Historical

Williams–Kilburn tube (1946–47) · Delay line memory (1947) · Mellon optical memory (1951) · Selectron tube (1952) · Dekatron · T-RAM (2009) · Z-RAM (2002–2010)

Non-volatile

ROM

Mask ROM · PROM · EPROM · EEPROM · **Flash memory**

NVRAM

ReRAM

Early stage NVRAM

FeRAM · MRAM · PCM (3D XPoint) · FeFET memory

Magnetic

Magnetic tape data storage (Linear Tape-Open) · Hard disk drive

Optical

Optical disc · 5D optical data storage

In development

CBRAM · Racetrack memory · NRAM · Millipede memory · ECRAM

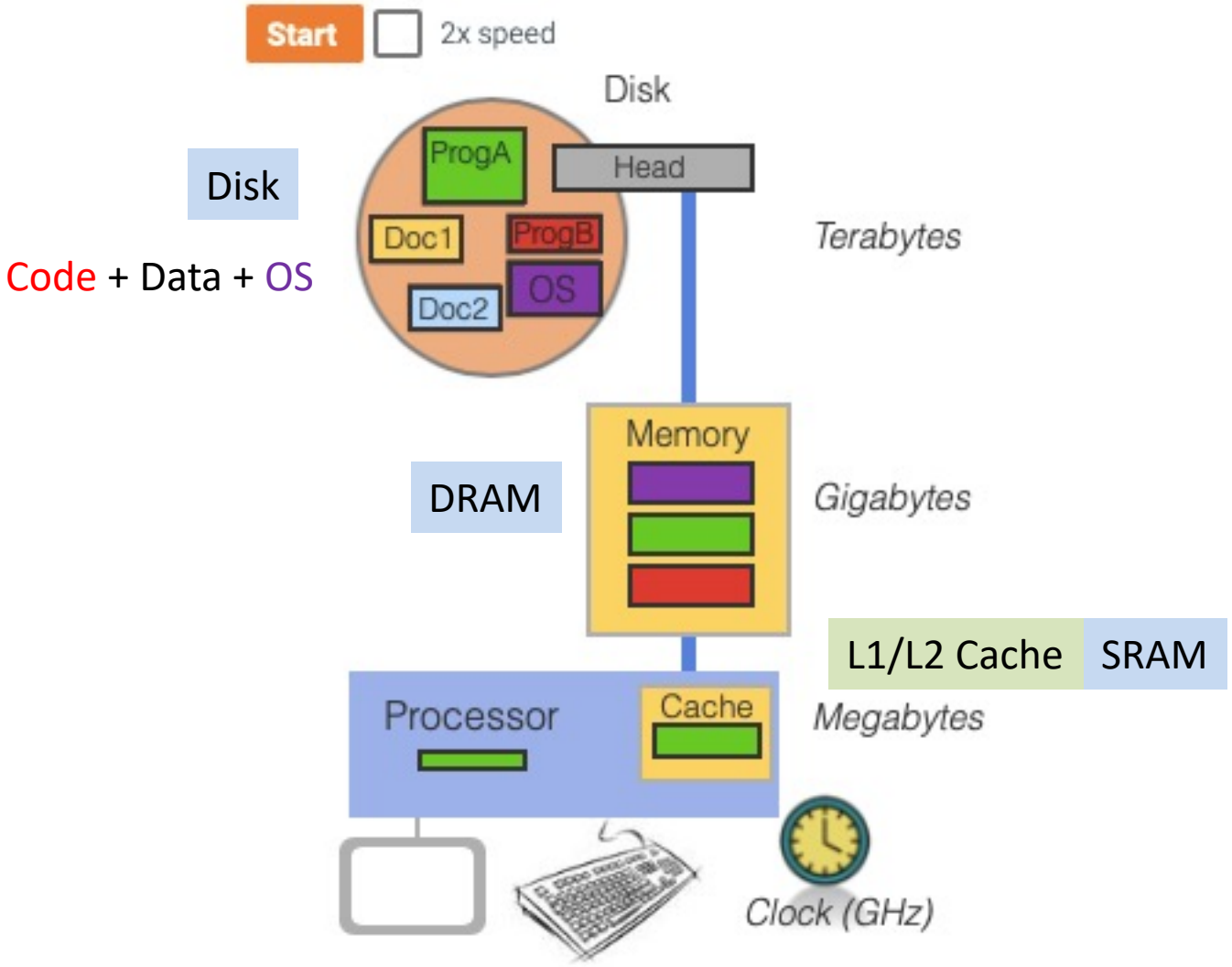
Historical

Paper data storage (1725) · Drum memory (1932) · Magnetic-core memory (1949) · Plated wire memory (1957) · Core rope memory (1960s) · Thin-film memory (1962) · Disk pack (1962) · Twistor memory (~1968) · Bubble memory (~1970) · Floppy disk (1971)

Computer Memory Org

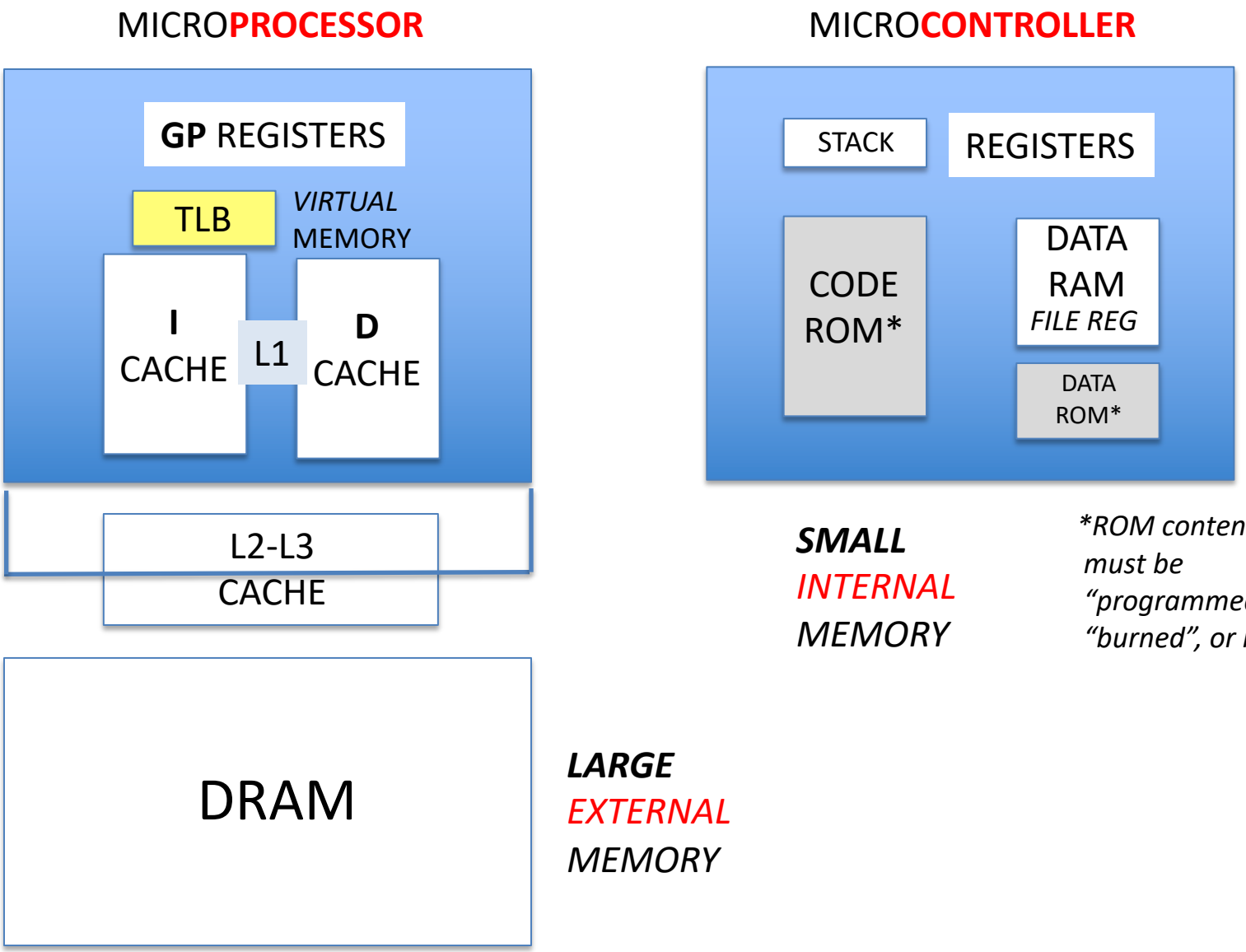
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

1.6.1: Some computer components.



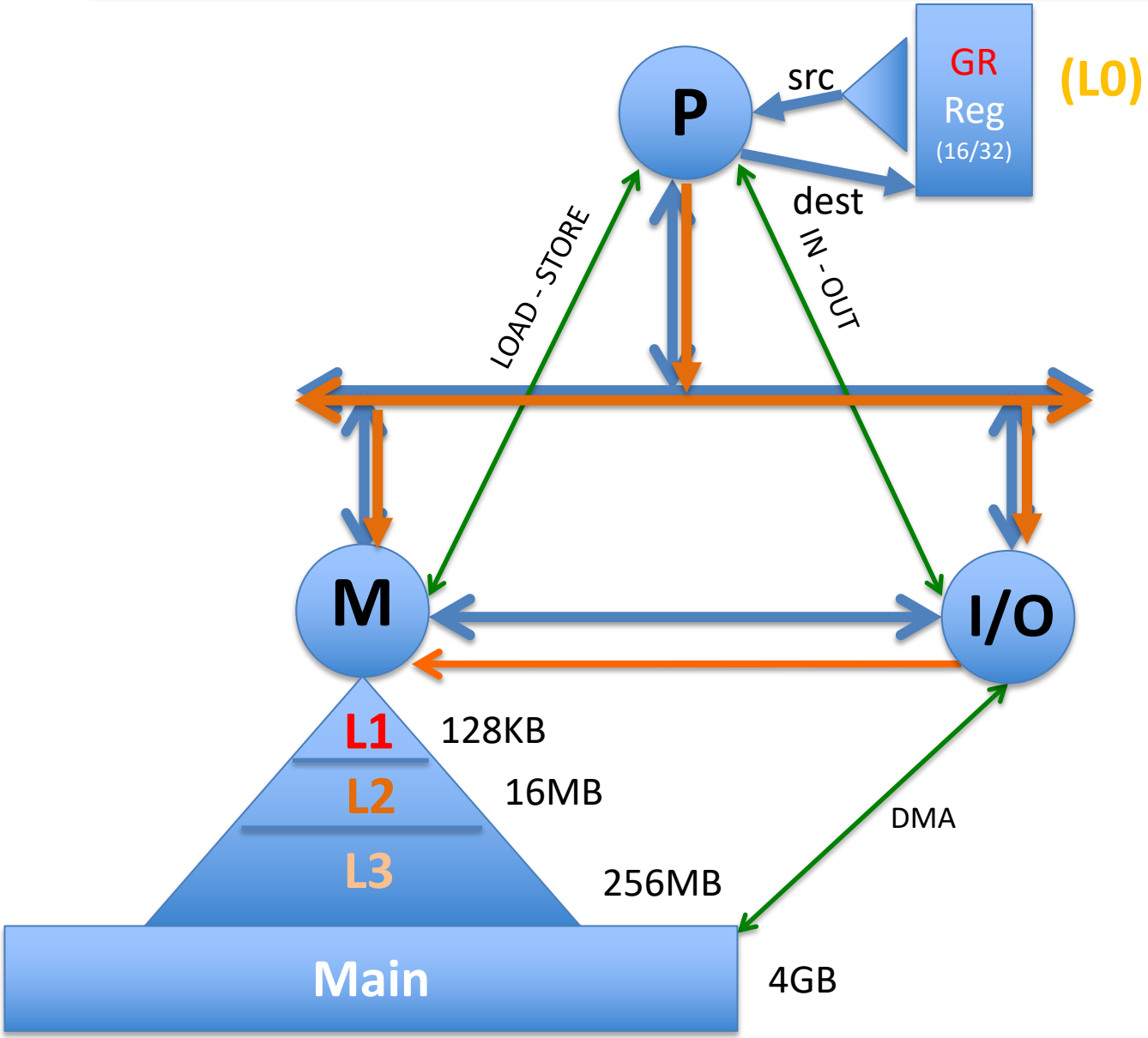
Memory Models

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



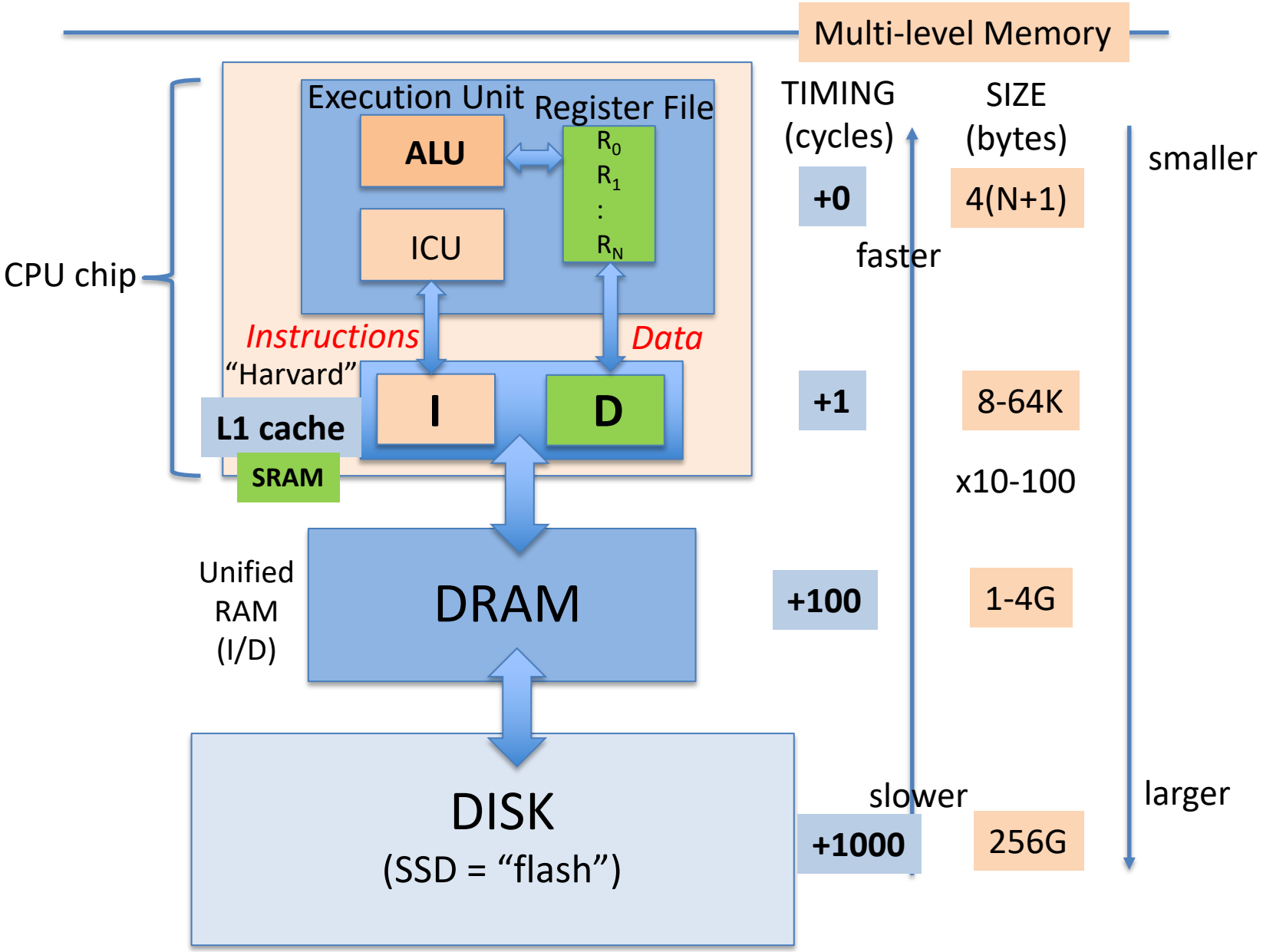
System Org: Multilevel Memory

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Broman
©2013-23



CPU Org + Memory Hierarchy

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



RAM/ROM (x8) Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

MCS-8

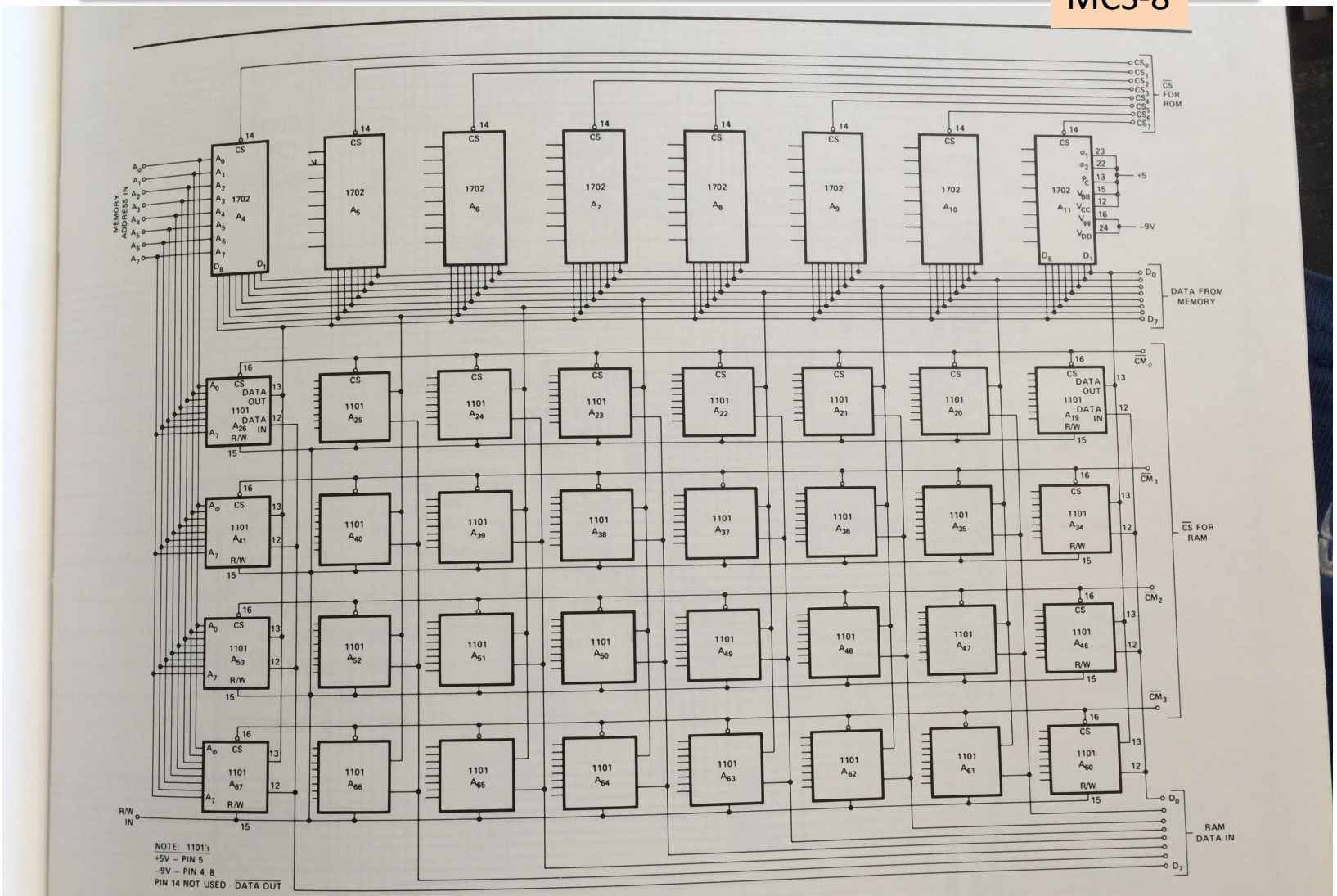


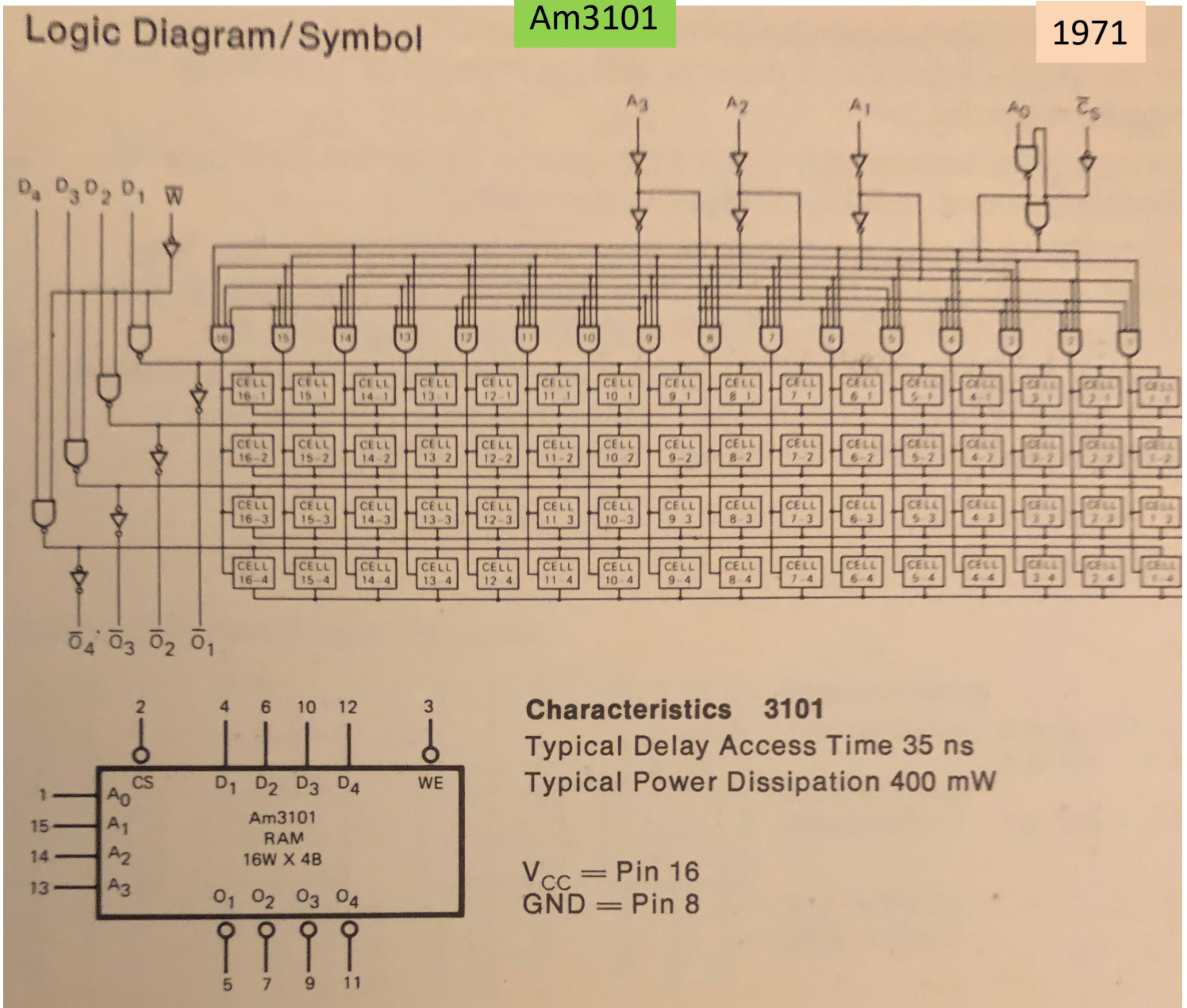
Figure 9. MCS-8 Memory System

AMD 64-Bit Bipolar SRAM

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Am3101

1971



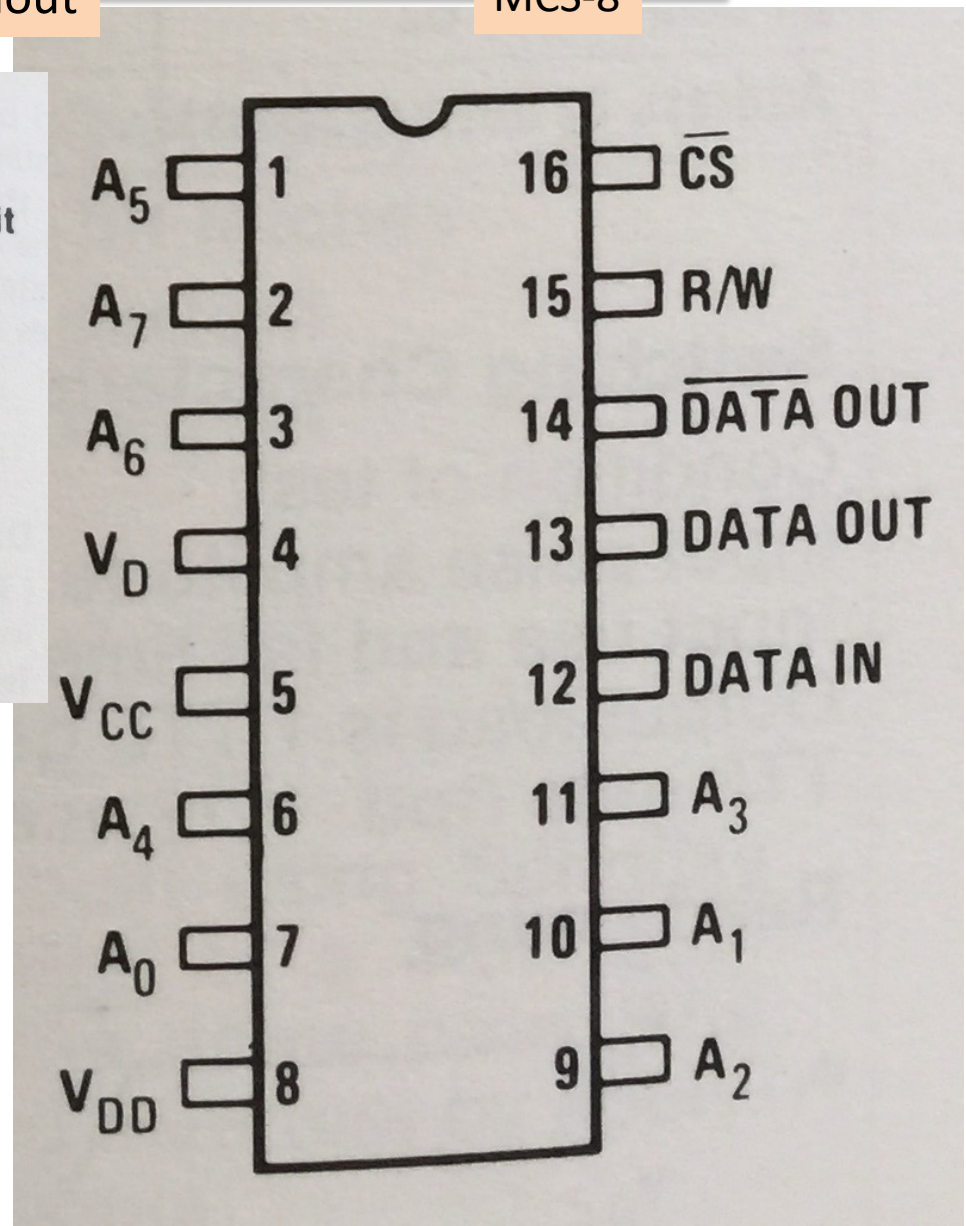
i1101A 256x1 SRAM

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Pinout

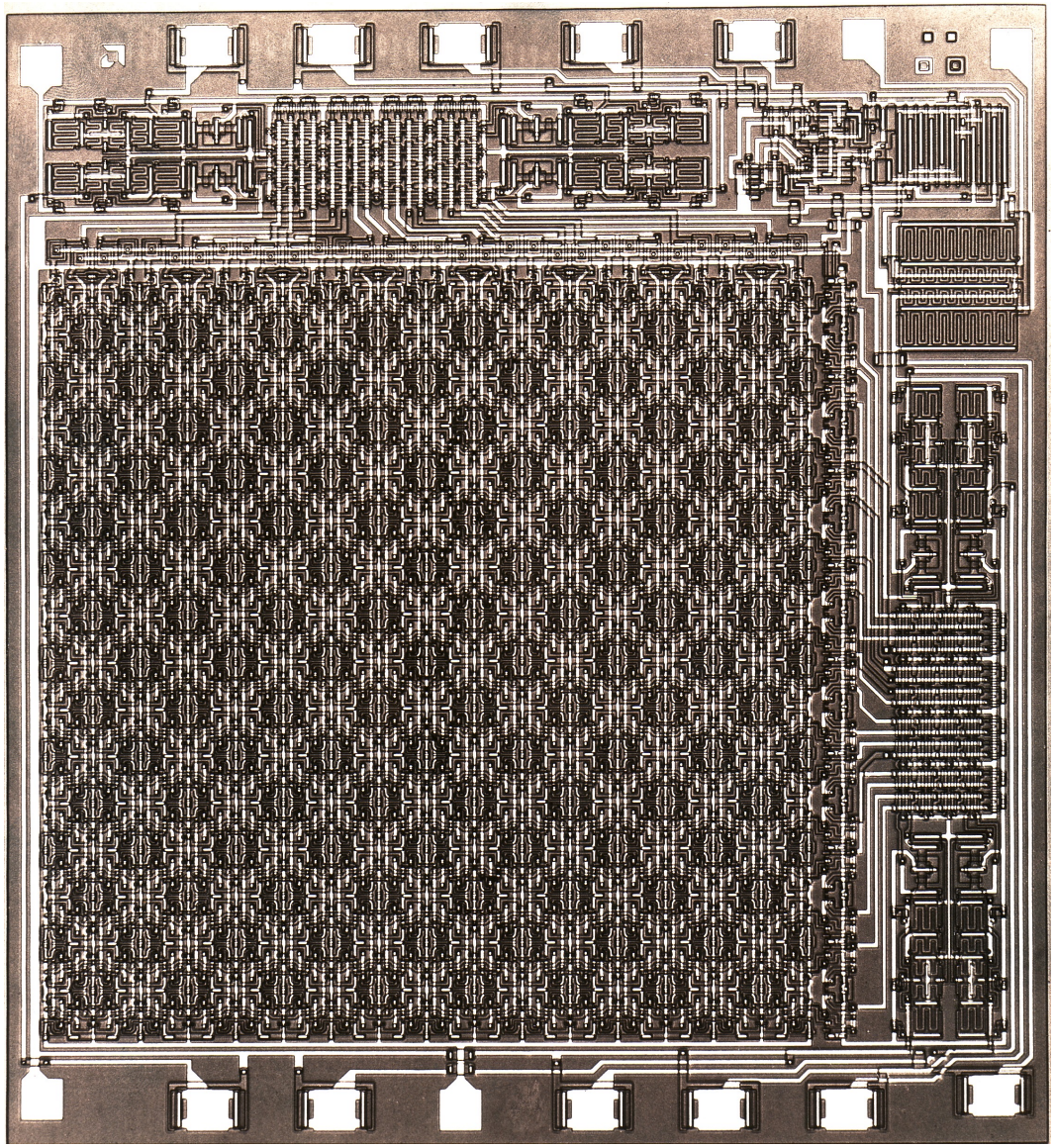
MCS-8

- Access time below 750 ns typically, 1.0 μsec maximum — 1101A1; 1.5 μsec maximum — 1101A: over temperature
- Low power dissipation—typically less than 1.5 mW/bit during access
- Low power standby mode
- Directly DTL and TTL compatible
- OR-Tie capability
- Simple memory expansion—chip-select input lead
- Fully decoded—on-chip address decode and sense
- Inputs protected against static charge
- Ceramic and plastic package
- Silicon gate MOS technology



Am1101A 256x1 SRAM

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

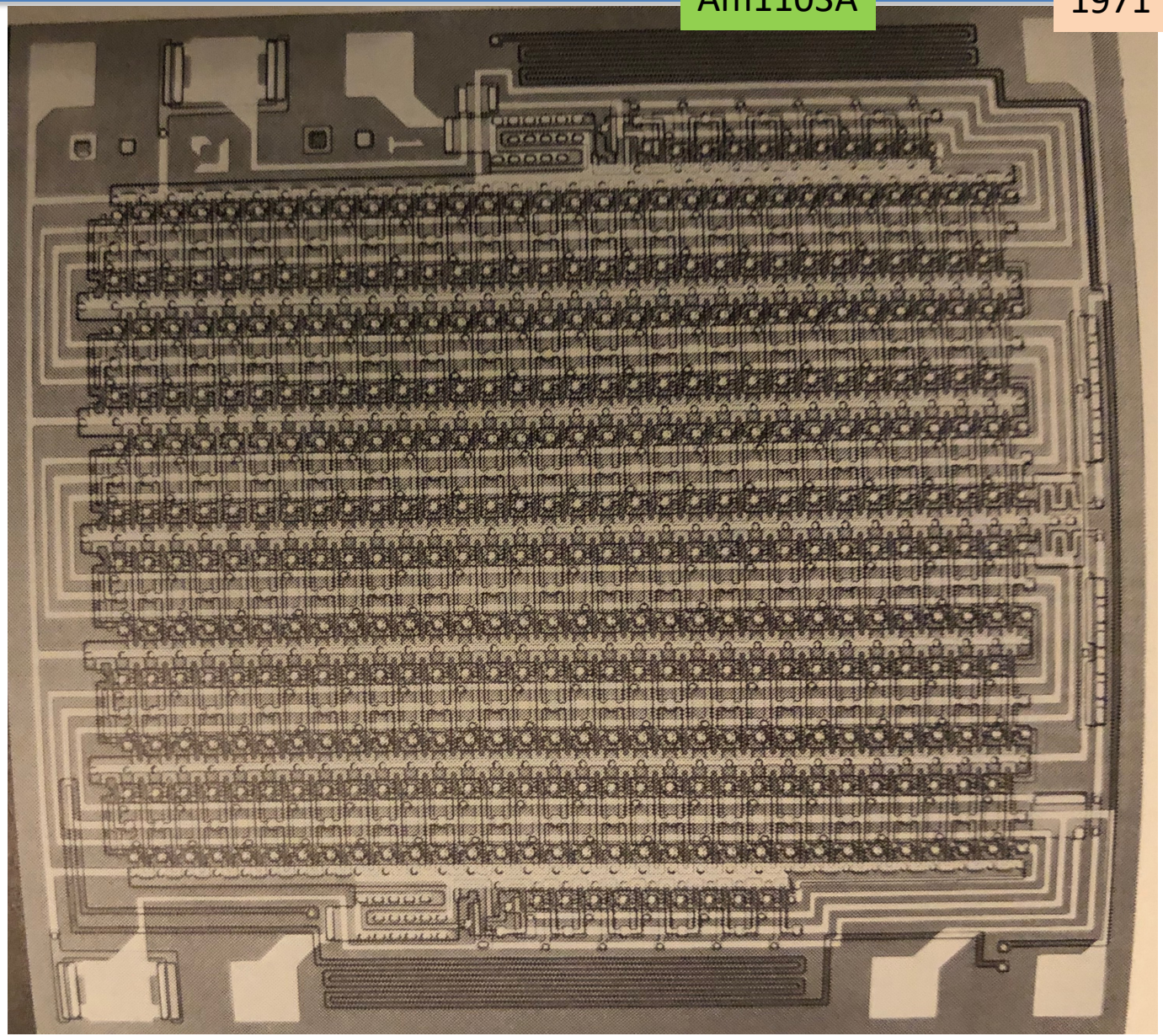


Am1103 1Kx1 DRAM

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Am1103A

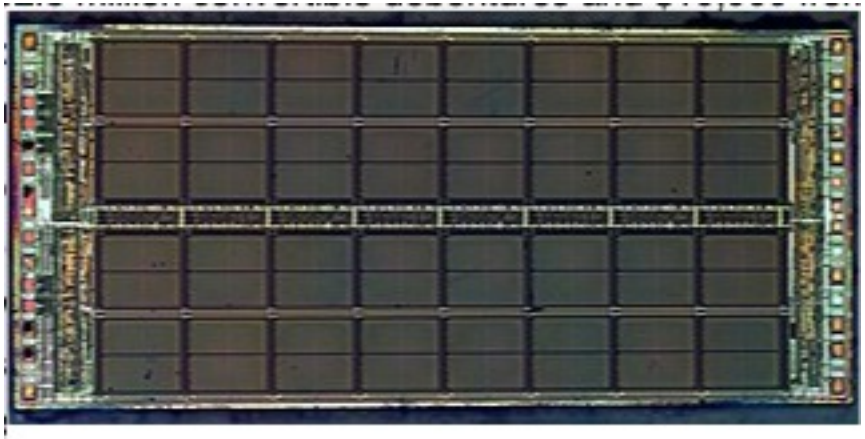
1971



Memory Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

DRAM 1T



Dynamic random-access memory (**DRAM**) is a type of random access semiconductor memory that stores each bit of data in a memory cell consisting of a tiny capacitor and a transistor, typically a MOSFET. The capacitor can either be charged or discharged; these two states are taken t

SRAM 4T

Static random-access memory is a type of semiconductor random-access memory (RAM) that uses bistable latching circuitry (flip-flop) to store each bit. SRAM exhibits data remanence, but it is still *volatile* in the conventional sense that data is eventually lost when the memory is not powered.

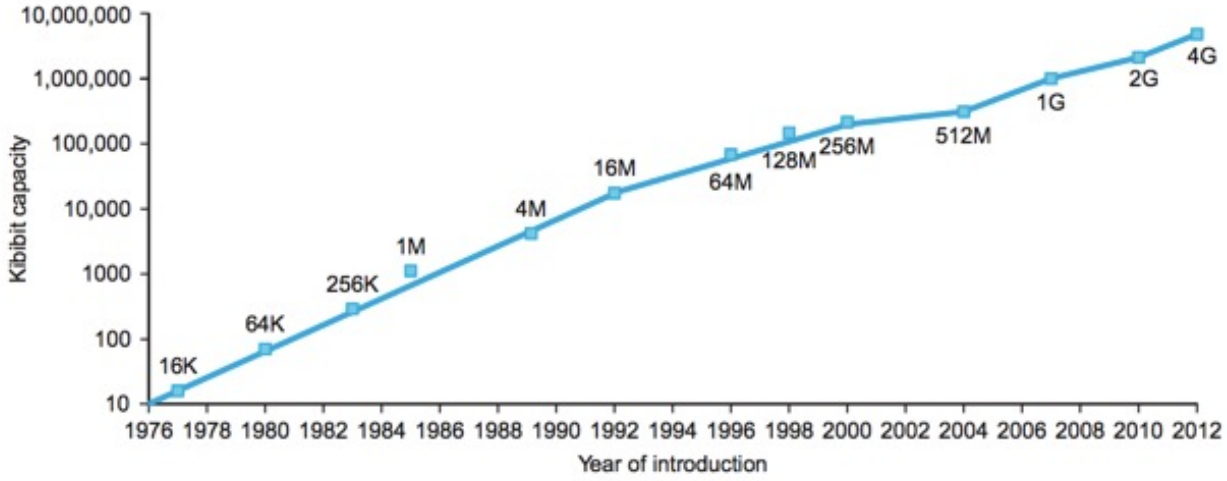
DRAM Timeline

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Patterson & Hennessy

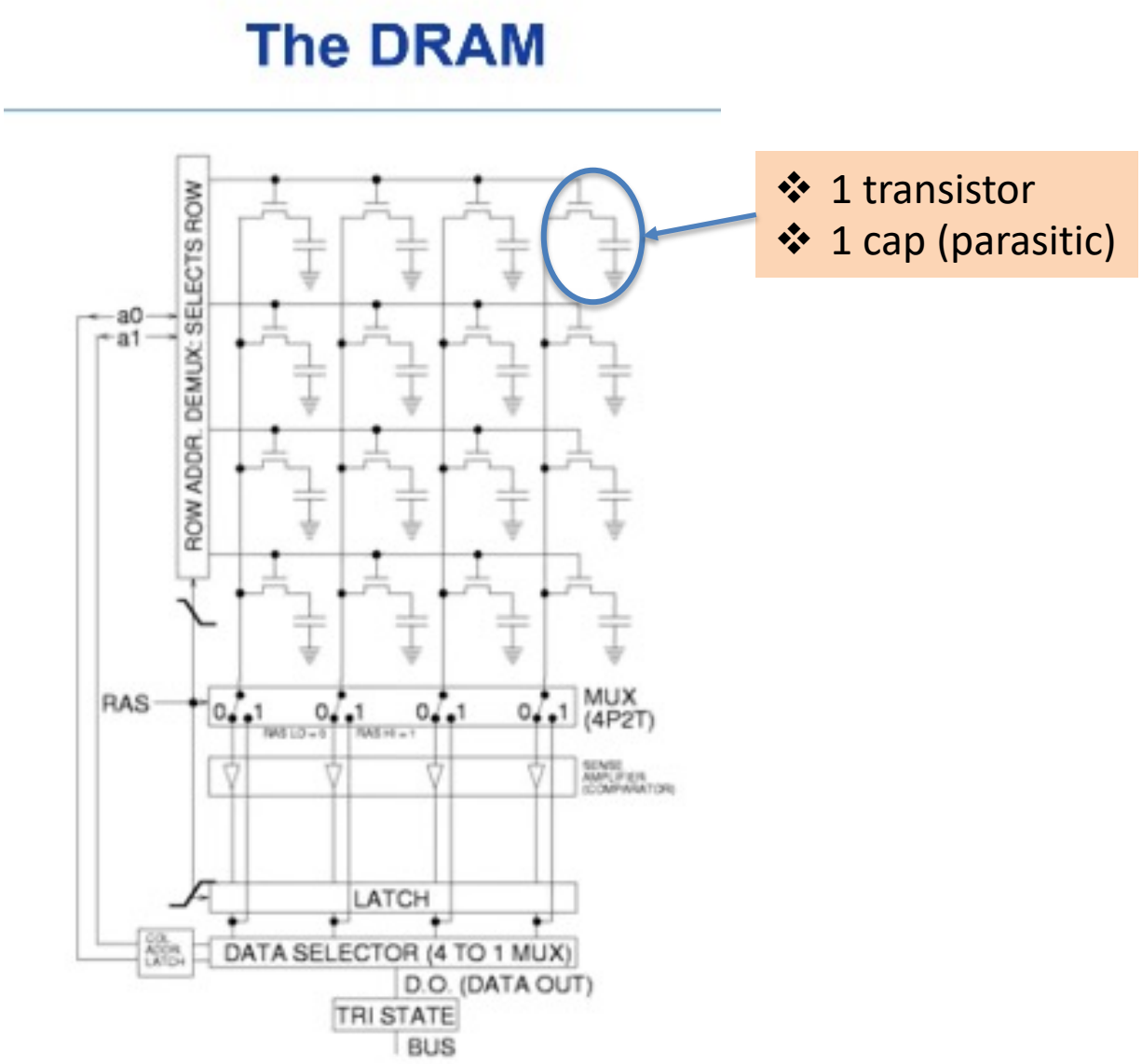
Figure 1.5.1: Growth of capacity per DRAM chip over time (COD Figure 1.11).

The y-axis is measured in kibibits (2^{10} bits). The DRAM industry quadrupled capacity almost every three years, a 60% increase per year, for 20 years. In recent years, the rate has slowed down and is somewhat closer to doubling every two years to three years.



DRAM Schematic

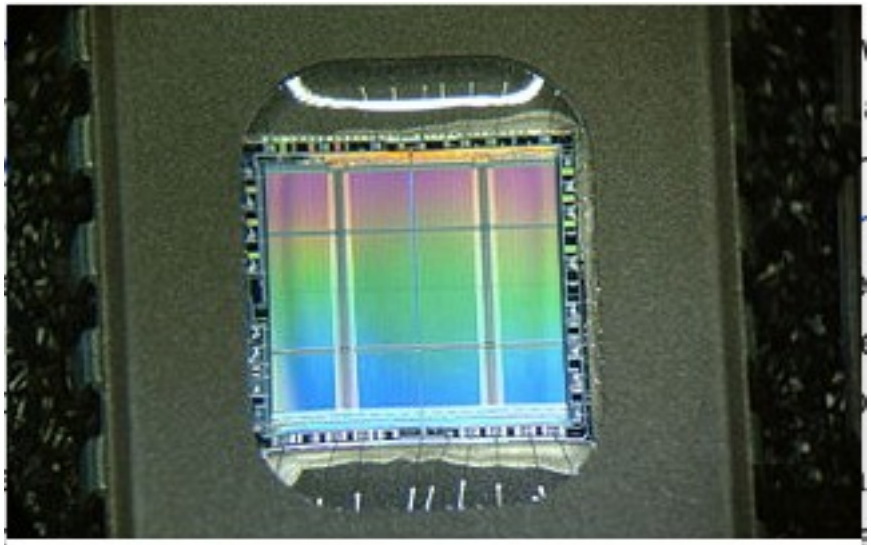
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



Memory Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

ROM



Read-only memory (**ROM**) is a type of non-volatile memory used in computers and other electronic devices. Data stored in ROM cannot be electronically modified after the manufacture of the memory device. Read-only memory is useful for storing software that is rarely changed during the life of the syst

- ROM (masked)
- PROM
- EPROM
- EEPROM
- **Flash E^2**

Flash ROM

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

WIKIPEDIA
The Free Encyclopedia

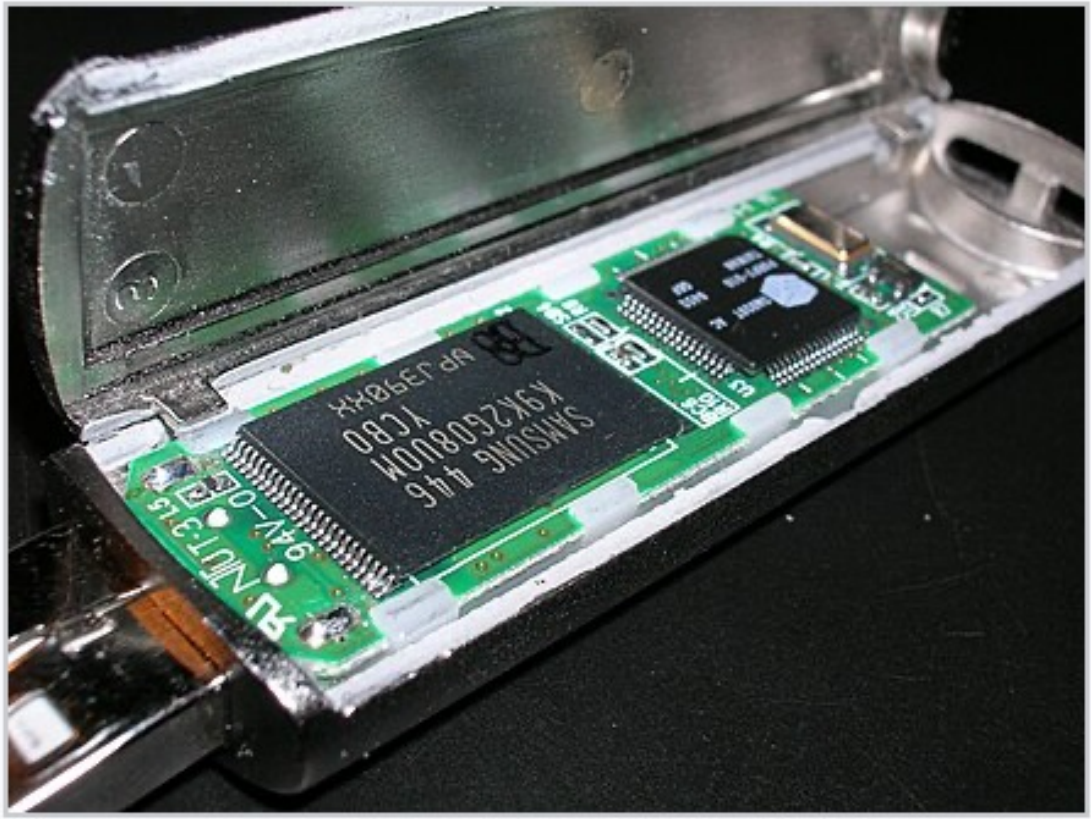
Flash memory is an electronic non-volatile computer memory storage medium that can be electrically erased and reprogrammed. The two main types of flash memory, **NOR flash** and **NAND flash**, are named for the **NOR** and **NAND** logic gates. Both use the same cell design, consisting of floating gate MOSFETs. They differ at the circuit level depending on whether the state of the bit line or word lines is pulled high or low: in NAND flash, the relationship between the bit line and the word lines resembles a NAND gate; in NOR flash, it resembles a NOR gate.

Flash memory, a type of floating-gate memory, was invented at Toshiba in 1980 and is based on EEPROM technology. Toshiba began marketing flash memory in 1987.[1] EPROMs had to be erased completely before they could be rewritten. NAND flash memory, however, may be erased, written, and read in blocks (or pages), which generally are much smaller than the entire device. NOR flash memory allows a single machine word to be written – to an erased location – or read independently. A flash memory device typically consists of one or more flash memory chips (each holding many flash memory cells), along with a separate flash memory controller chip.

Flash ROM

WIKIPEDIA
The Free Encyclopedia

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



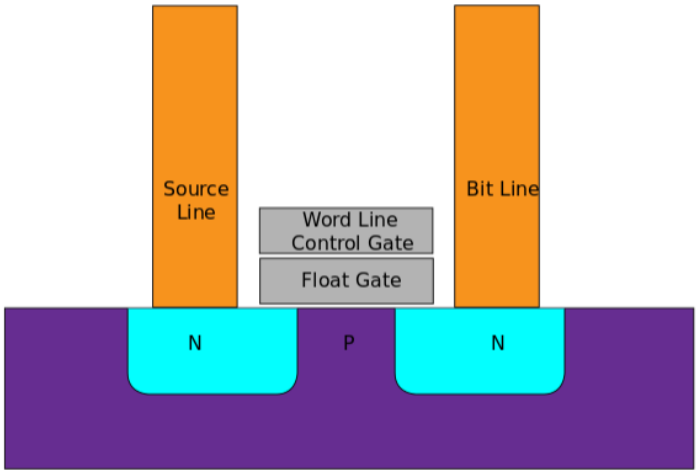
A disassembled USB flash drive. The chip on the left is flash memory. The controller is on the right.

Flash ROM

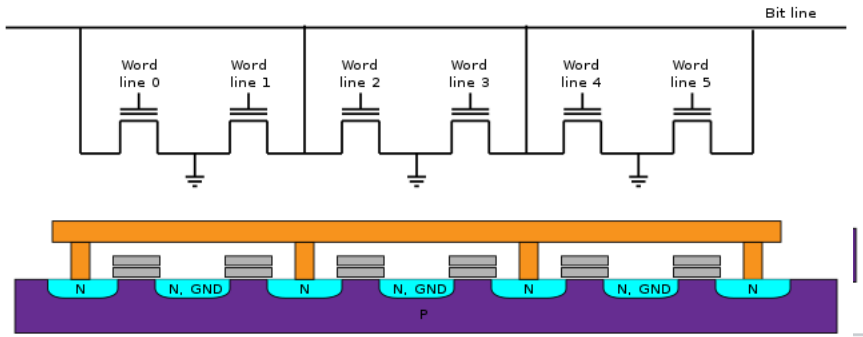
DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

WIKIPEDIA
The Free Encyclopedia

- Floating Gate

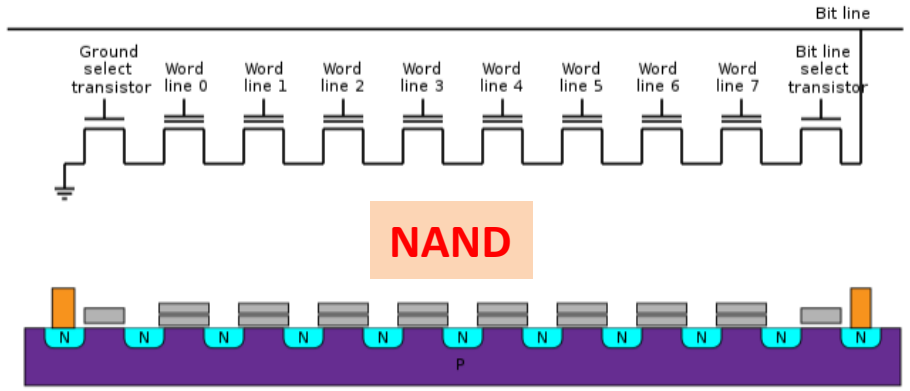


A flash memory cell



NOR flash memory wiring and structure on silicon

NOR



NAND flash memory wiring and structure on silicon

Flash ROM

WIKIPEDIA
The Free Encyclopedia

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Flash memory stores information in an array of memory cells made from floating-gate transistors

- In single-level cell (**SLC**) devices, each cell stores only one bit of information.
- Multi-level cell (**MLC**) devices, including triple-level cell (**TLC**) devices, can store more than one bit per cell.

4 levels → 2 bits

2x DRAM density

Flash ROM

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

WIKIPEDIA
The Free Encyclopedia

NAND

NAND memories

NAND flash architecture was introduced by Toshiba in 1989.[97] These memories are accessed much like block devices, such as hard disks. Each block consists of a number of pages. The pages are typically 512,[98] 2,048 or 4,096 bytes in size. Associated with each page are a few bytes (typically 1/32 of the data size) that can be used for storage of an error correcting code (**ECC**) checksum.

Typical block sizes include:

- 32 pages of 512+16 bytes each for a block size (effective) of **16** KiB
- 64 pages of 2,048+64 bytes each for a block size of **128** KiB[99]
- 64 pages of 4,096+128 bytes each for a block size of **256** KiB[100]
- 128 pages of 4,096+128 bytes each for a block size of **512** KiB.

While reading and programming is performed on a page basis, erasure can only be performed on a block basis

WOM!

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

April 1, 1980

signetics FULLY ENCODED, 9046 x N, RANDOM ACCESS WRITE-ONLY-MEMORY

25120

Do Not Copy

FINAL SPECIFICATION(10)

DESCRIPTION

The Signetics 25000 Series 9046XN Random Access Write-Only-Memory employs both enhancement and depletion mode P-Channel, N-Channel, and neu(1) channel MOS devices. Although a static device, a single TTL level clock phase is required to drive the on-board multi-port clock generator. Data refresh is accomplished during CB and LH periods(11). Quadri-state outputs (when applicable) allow expansion in many directions, depending on organization.

The static memory cells are operated dynamically to yield extremely low power dissipation. All inputs and outputs are directly TTL compatible when proper interfacing circuitry is employed.

Device construction is more or less S.O.S.(2).

FEATURES

- FULLY ENCODED MULTI-PORT ADDRESSING
- WRITE CYCLE TIME 80nS (MAX. TYPICAL)
- WRITE ACCESS TIME(3)
- POWER DISSIPATION 10uW/BIT TYPICAL
- CELL REFRESH TIME 2mS (MIN. TYPICAL)

BIPOLAR COMPATIBILITY

All data and clock inputs plus applicable outputs will interface directly or nearly directly with bipolar circuits of suitable characteristics. In any event use 1 amp fuses in all power supply and data lines.

INPUT PROTECTION

All terminals are provided with slip-on latex protectors for the prevention of Voltage Destruction. (PILL packaged devices do not require protection).

SILICON PACKAGING

Low cost silicon DIP packaging is implemented and reliability is assured by the use of a non-hermetic sealing technique which prevents the entrapment of harmful ions, but which allows the free exchange of friendly ions.

SPECIAL FEATURES

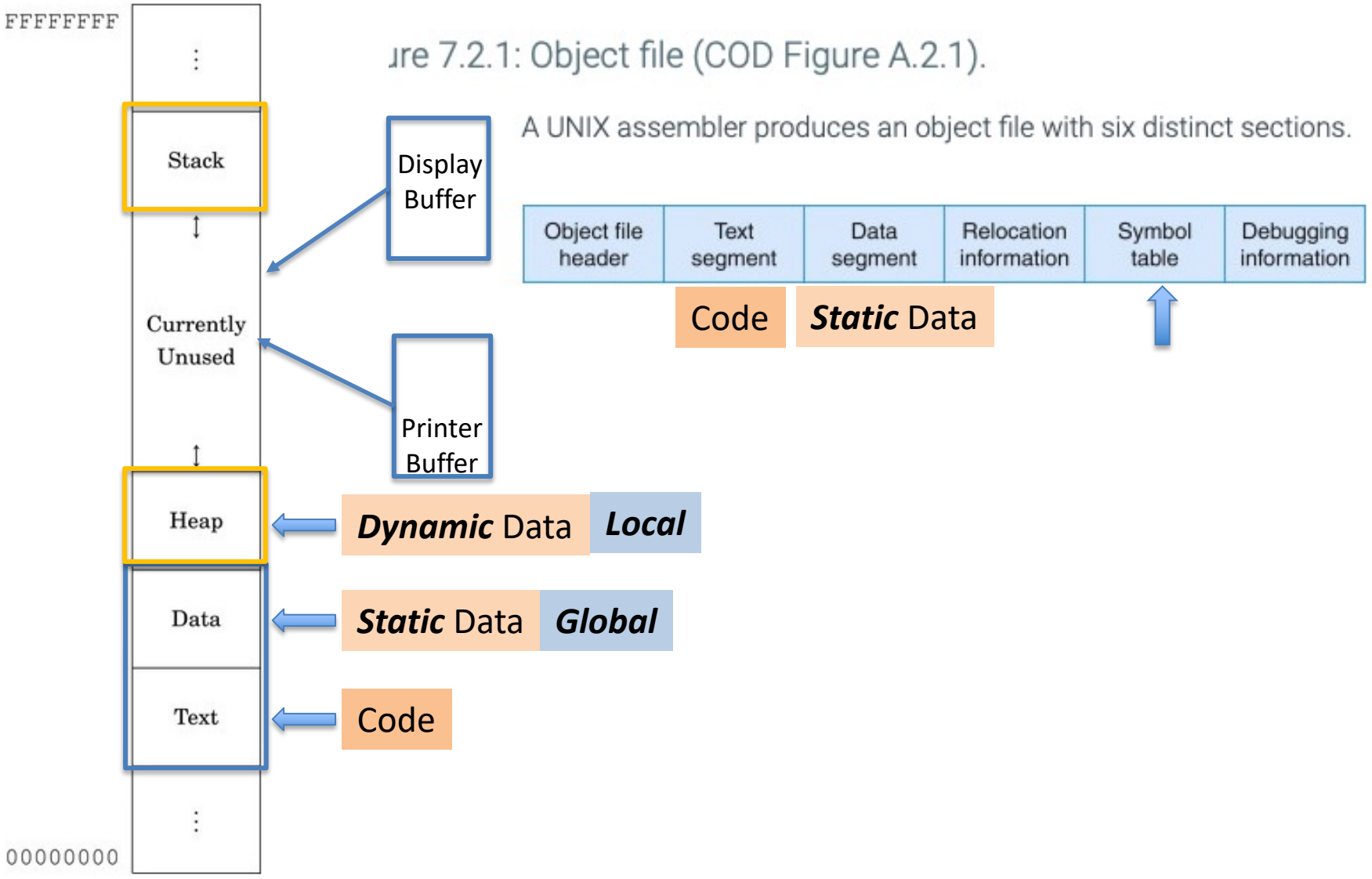
Because of the employment of the Signetics' proprietary Sanderson-Rabbet Channel the 25120 will provide 50% higher speed than you will obtain.

COOLING

The 25120 is easily cooled by employment of a six-foot

Memory Segments

Jeff Drobman
©2016-23



ure 7.2.1: Object file (COD Figure A.2.1).

A UNIX assembler produces an object file with six distinct sections.

Object file header	Text segment	Data segment	Relocation information	Symbol table	Debugging information

Code **Static** Data

Typical memory layout for a program with a 32-bit address space.

GiB/TiB ($2^{30}/2^{40}$)

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

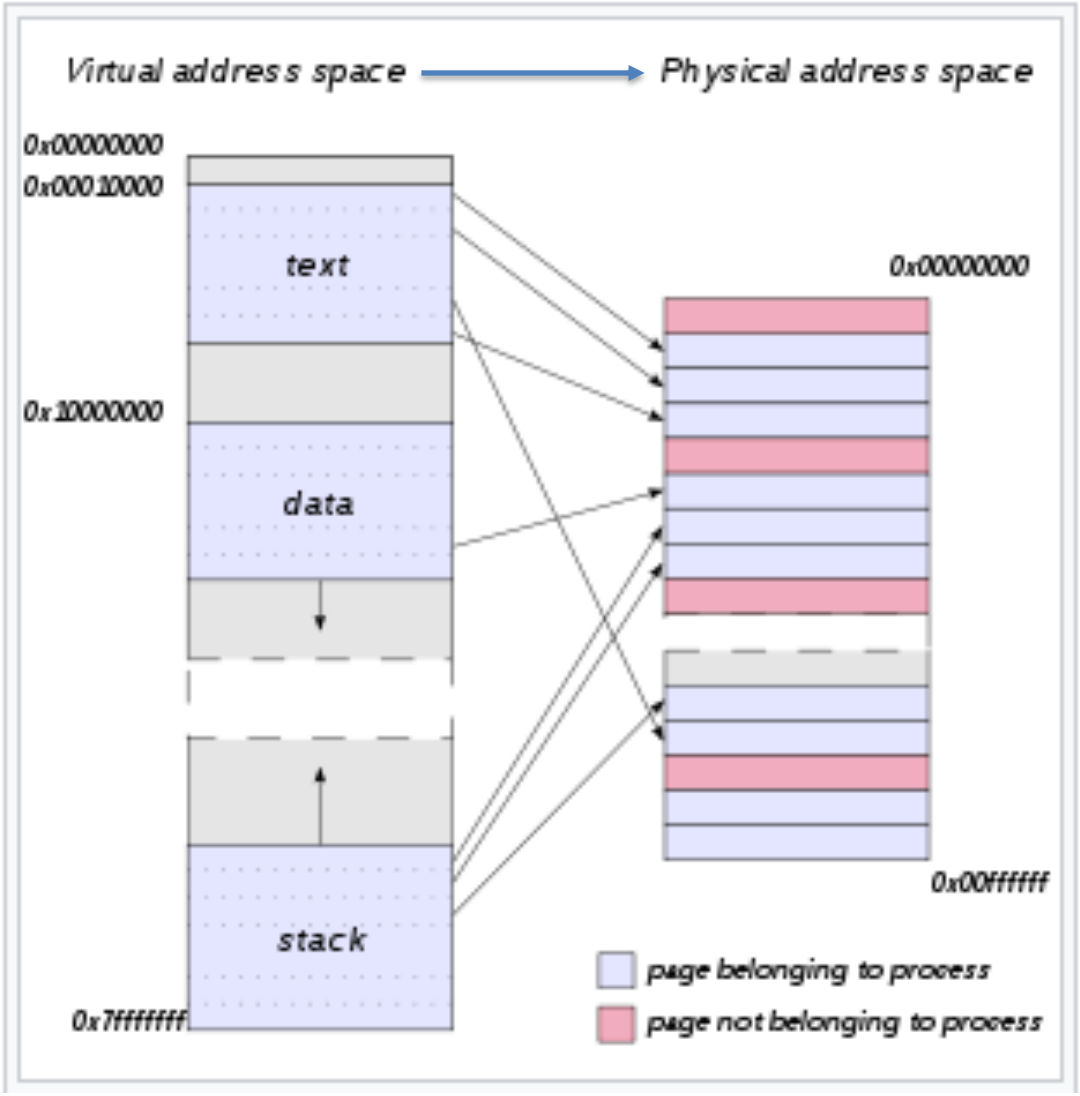
Decimal	Abbreviation	Value	Binary term	Abbreviation	Value	% Larger
kilobyte	KB	10^3	kibibyte	KiB	2^{10}	2%
megabyte	MB	10^6	mebibyte	MiB	2^{20}	5%
gigabyte	GB	10^9	gibibyte	GiB	2^{30}	7%
terabyte	TB	10^{12}	tebibyte	TiB	2^{40}	10%
petabyte	PB	10^{15}	pebibyte	PiB	2^{50}	13%
exabyte	EB	10^{18}	exbibyte	EiB	2^{60}	15%
zettabyte	ZB	10^{21}	zebibyte	ZiB	2^{70}	18%
yottabyte	YB	10^{24}	yobibyte	YiB	2^{80}	

Actual
1024
1,048,576
$1.074x10^9$
$1.0995x10^{12}$

Ordinal	Power of 2	Power of 10	Actual
1K	2^{10}	10^3	1024
1M	2^{20}	10^6	1,048,576
1G	2^{30}	10^9	$1.074x10^9$
1T	2^{40}	10^{12}	$1.0995x10^{12}$

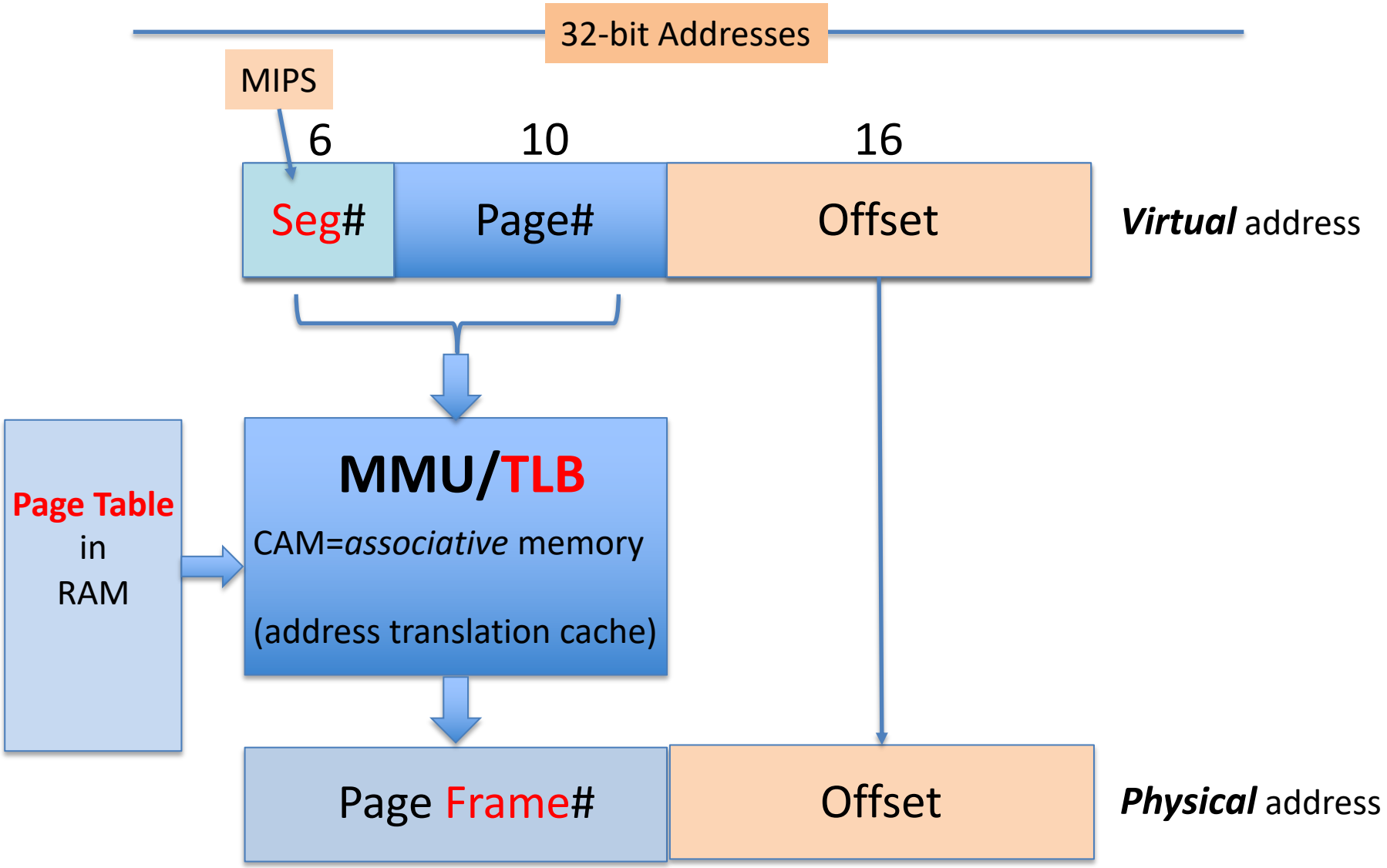
Virtual Memory

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



Virtual Memory

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



Chips

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Logic
&
Bit-Slice
Microprocessors

Logic IC's: ALU *Slices*

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

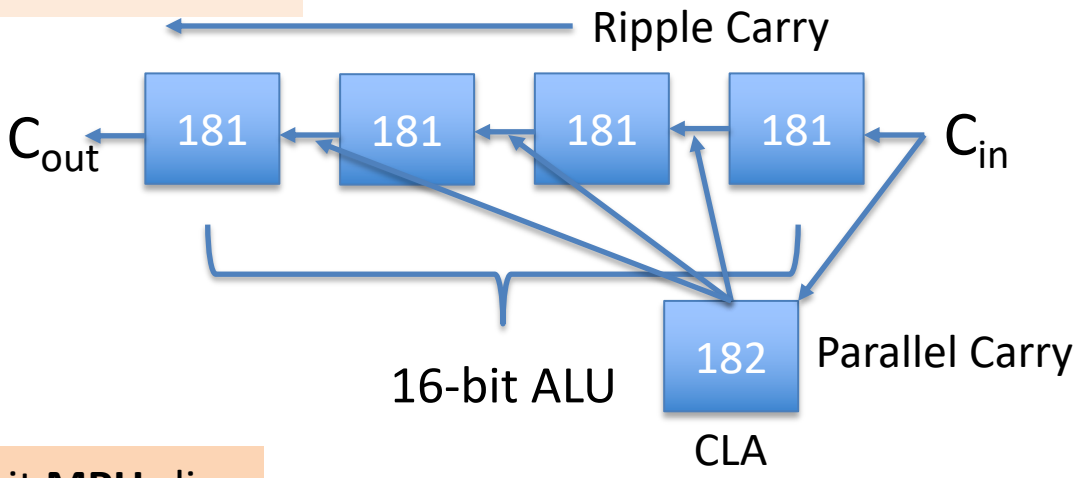
Bit-slice 1965-75

- Fairchild 9300 series
- Signetics
- National
- Motorola
- Texas Instruments

Texas Instruments

- 54/7400 Series
  - 54/74SN**181**
    - 4-bit **ALU** slice

- 54 → **Mil** temp
- 74 → **Com'l** temp



Am2505 2x4-bit **Multiplier** slice

- 4-bit **MPU** slice

Replaced by **Am2900** *family* → Am29**01** ALU + Register file *microprogrammed*
Am29**10** *Microprogram sequencer*
Am29**02** CLA

Am2505 Multiplier

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Bit-slice 1971-80

#18 SN 7437 N

LO4
N7400A Sig.
DM7400N Nat.
NL7400K M
UC7400 F

Am2505

Four-Bit by Two-Bit 2's Complement Multiplier
Advanced Micro Devices
Complex Digital Integrated Circuits

Distinctive Characteristics:

- Provides 2's complement multiplication at high speed without correction.
- Can be used in an iterative scheme or time sequenced mode.
- Multiplies two 12-bit signed numbers in typically 200ns.
- Multiplies in active HIGH (positive logic) or active LOW (negative logic) representations.
- Easy correction for unsigned, sign-magnitude or 1's complement multiplication.
- 100% reliability assurance testing in compliance with MIL STD 883.

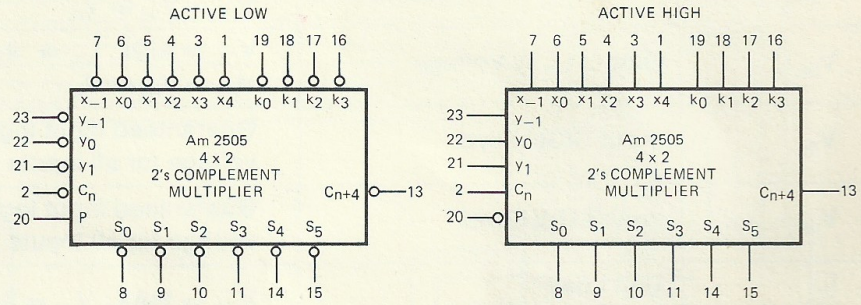
FUNCTIONAL DESCRIPTION:

The Am2505 is a high-speed digital multiplier that can multiply numbers represented in the 2's complement notation and produce a 2's complement product without correction. The device consists of a 4x2 multiplier that can be connected to form iterative arrays able to multiply numbers either directly, or in a time sequenced arrangement. The device assumes that the most significant digit in a word carries a negative weight, and can therefore be used in arrays where the multiplicand and multiplier have different word lengths. The multiplier uses the quaternary algorithm and performs the function $S = XY + K$ where K is the input field used to add partial products generated in the array. At the beginning of the array the K inputs are available to add a signed constant to the least significant part of the product. Multiplication of an m bit number by an n bit number in an array results in a product having m+n bits so that all possible combinations of product are accounted for. If a conventional 2's complement product is required the most significant bit can be ignored, and overflow conditions can be detected by comparing the last two product digits.

Figure 2 shows how multipliers are connected together in an array. A number of connection schemes are possible. Figure 4 shows diagramatically the connection scheme that results in the fastest multiply. If higher speed is required an array can be split into several parts, and the parts added with high-speed look-ahead carry adders such as the Am9340.

Provision is made in the design for multiplication in the active high (positive logic) or active low (negative logic) representations simply by reinterpreting the active level of the input operands, the product, and a polarity control $\overline{P}$. For a more complete description and applications the user is referred to the Am2505 Application Note.

LOGIC SYMBOLS

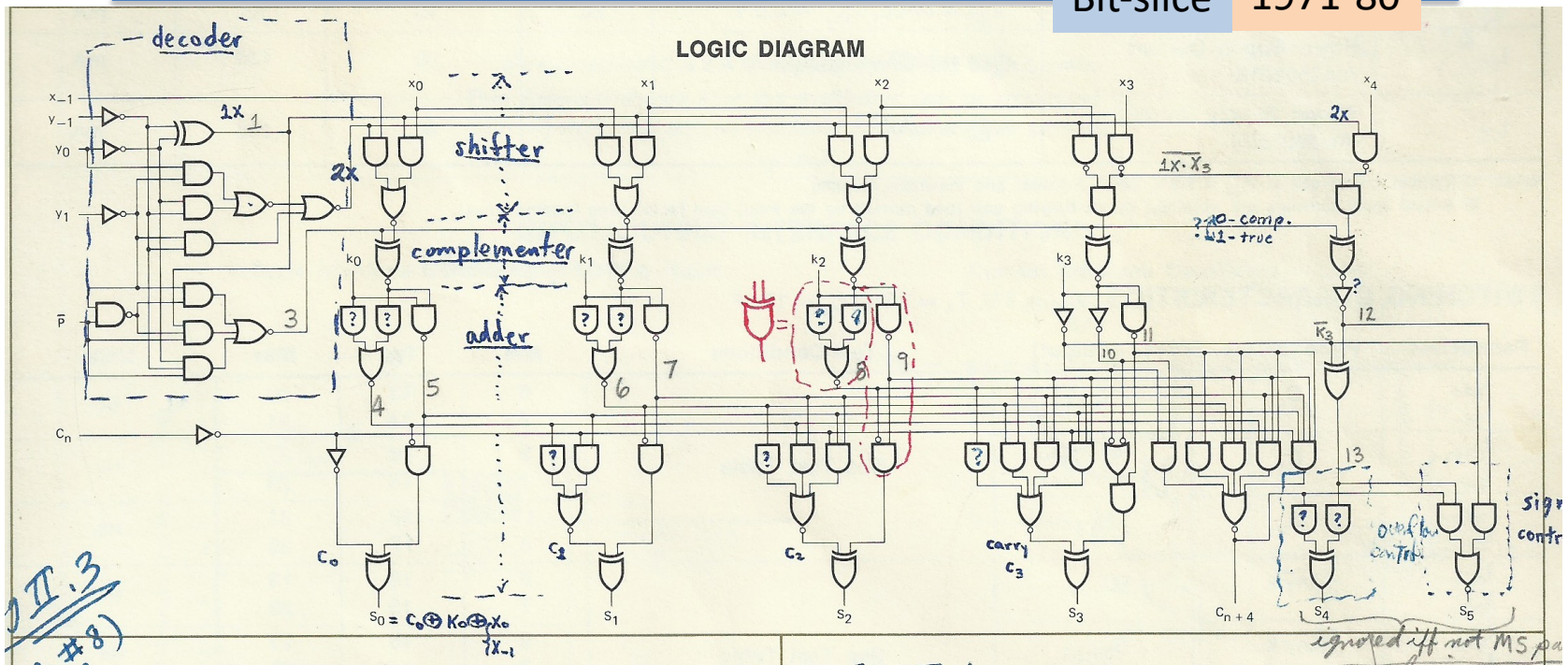


V_{CC} = PIN 24
GND = PIN 12

Am2505 Multiplier

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Bit-slice 1971-80

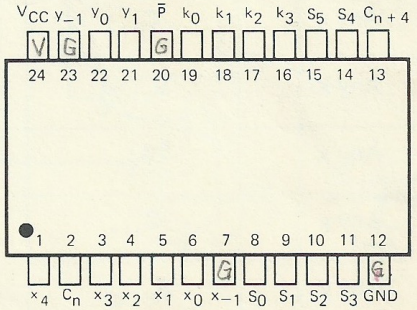


Am2505 ORDERING INFORMATION

Package Type	Temperature Range	Order Number
Silicone DIP	0°C to +75°C	AM250559C
Hermetic DIP	0°C to +75°C	AM250559F
Hermetic DIP	−55°C to +125°C	AM250551F
Hermetic Flat Pak	−55°C to +125°C	AM250551P
Dice	Note	AM2505XXD

Note: The dice supplied will contain units which meet both 0°C to +75°C and −55°C to +125°C temperature ranges.

Fig. II. 1. **CONNECTION DIAGRAM**
Top View



NOTE: PIN 1 is marked for orientation.

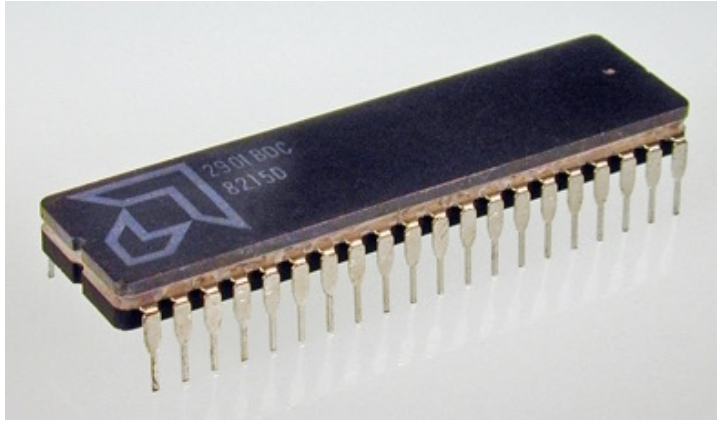
Am2900 Family

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Bit-slice 1975-85

AMD 2901 bit-slice processor family includes 2901 and 2903 4-bit microprocessors slices, 2909 and 2911 microprogram sequencers, 2910 microprogram controller and other support chips. The 2901 processor consists of 16 4-bit registers, 4-bit ALU and associated decoding/multiplexing circuits. The ALU accepts 9-bit microinstructions that specify source operands, ALU function and the destination register. The 2901 ALU can perform 8 different functions (they are encoded into 3 bits within the microinstruction): addition, subtraction and logic operations. Multiple 2901 bit-slice processors could be combined together to build microprocessors with any data width (in 4 bits increments).

Enhanced version of 2901, AMD 2903 has 9 new special ALU functions used for implementation of multiplication, division and normalization operations. The number of arithmetic and logic ALU functions in 2903 was increased to 15.



AMD Am2903: 4-bit-slice ALU

Am2900 Family

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Bit-slice 1975-85

Members of the Am2900 family [edit]

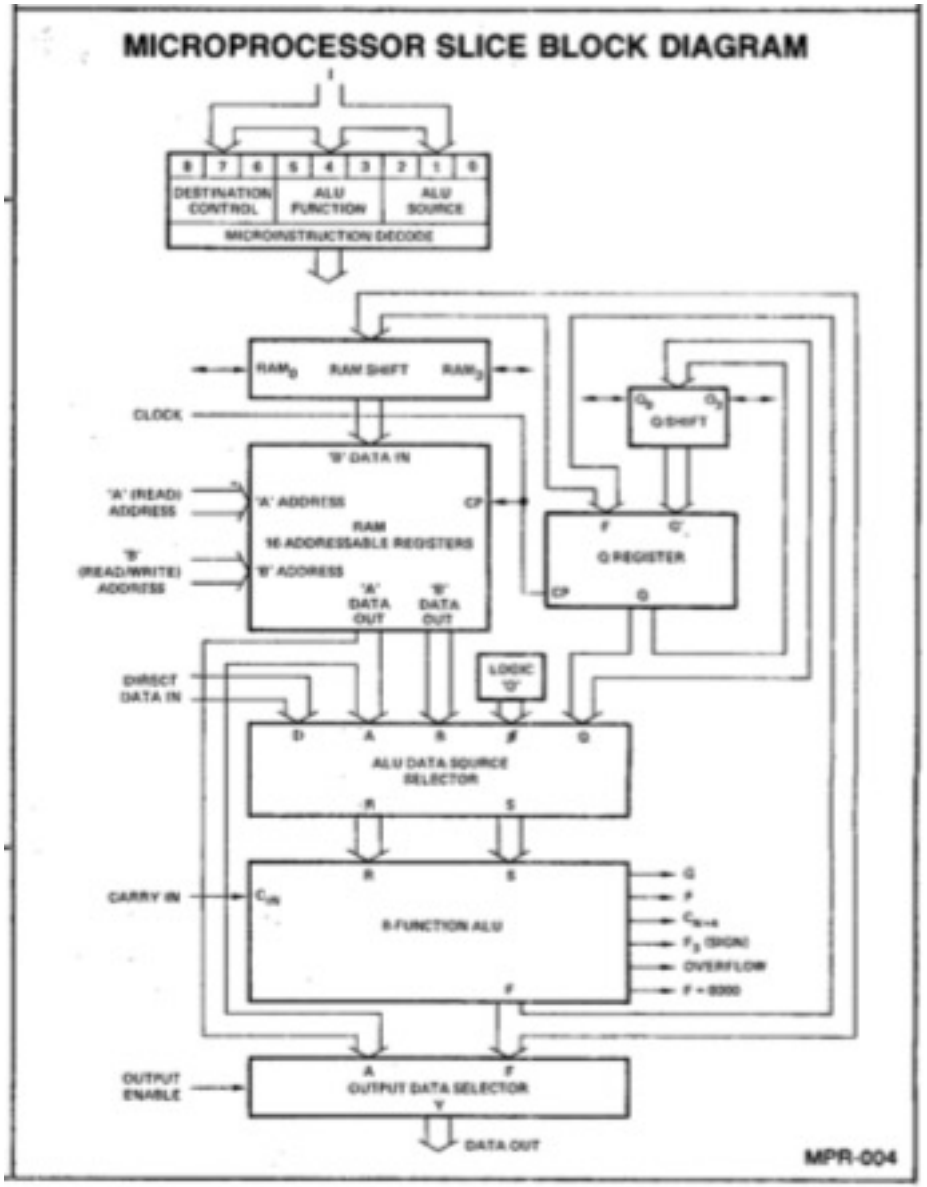
The Am2900 Family Data Book lists:[22]

- Am2901 – 4-bit bit-slice ALU (1975)
- Am2902 – Look-Ahead Carry Generator
- Am2903 – 4-bit-slice ALU, with hardware multiply
- Am2904 – Status and Shift Control Unit
- Am2905 – Bus Transceiver
- Am2906 – Bus Transceiver with Parity
- Am2907 – Bus Transceiver with Parity
- Am2908 – Bus Transceiver with Parity
- Am2909 – 4-bit-slice address sequencer
- Am2910 – 12-bit address sequencer
- Am2911 – 4-bit-slice address sequencer
- Am2912 – Bus Transceiver
- Am2913 – Priority Interrupt Expander
- Am2914 – Priority Interrupt Controller
- Am2915 – Quad 3-State Bus Transceiver
- Am2916 – Quad 3-State Bus Transceiver
- Am2917 – Quad 3-State Bus Transceiver
- Am2918 – Instruction Register, Quad D Register
- Am2919 – Instruction Register, Quad Register
- Am2920 – Octal D-Type Flip-Flop
- Am2921 – 1-to-8 Decoder
- Am2922 – 8-Input Multiplexer (MUX)
- Am2923 – 8-Input MUX
- Am2924 – 3-Line to 8-Line Decoder
- Am2925 – System Clock Generator and Driver
- Am2926 – Schottky 3-State Quad Bus Driver
- Am2927/Am2928 – Quad 3-State Bus Transceiver
- Am2929 – Schottky 3-State Quad Bus Driver
- Am2930 – Main Memory Program Control
- Am2932 – Main Memory Program Control
- Am2940 – Direct Memory Addressing (DMA) Generator
- Am2942 – Programmable Timer/Counter/DMA Generator
- Am2946/Am2947 – Octal 3-State Bidirectional Bus Transceiver
- Am2948/Am2949 – Octal 3-State Bidirectional Bus Transceiver
- Am2950/Am2951 – 8-bit Bidirectional I/O Ports
- Am2954/Am2955 – Octal Registers
- Am2956/Am2957 – Octal Latches
- Am2958/Am2959 – Octal Buffers/Line Drivers/Line Receivers
- Am2960 – Cascadable 16-bit Error Detection and Correction Unit
- Am2961/Am2962 – 4-bit Error Correction Multiple Bus Buffers
- Am2964 – Dynamic Memory Controller
- Am2965/Am2966 – Octal Dynamic Memory Driver

2901 Block Diagram

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Am2900



16-bit CPU = 4x 2901

Jeff Drobman
©2016-23

Am2900

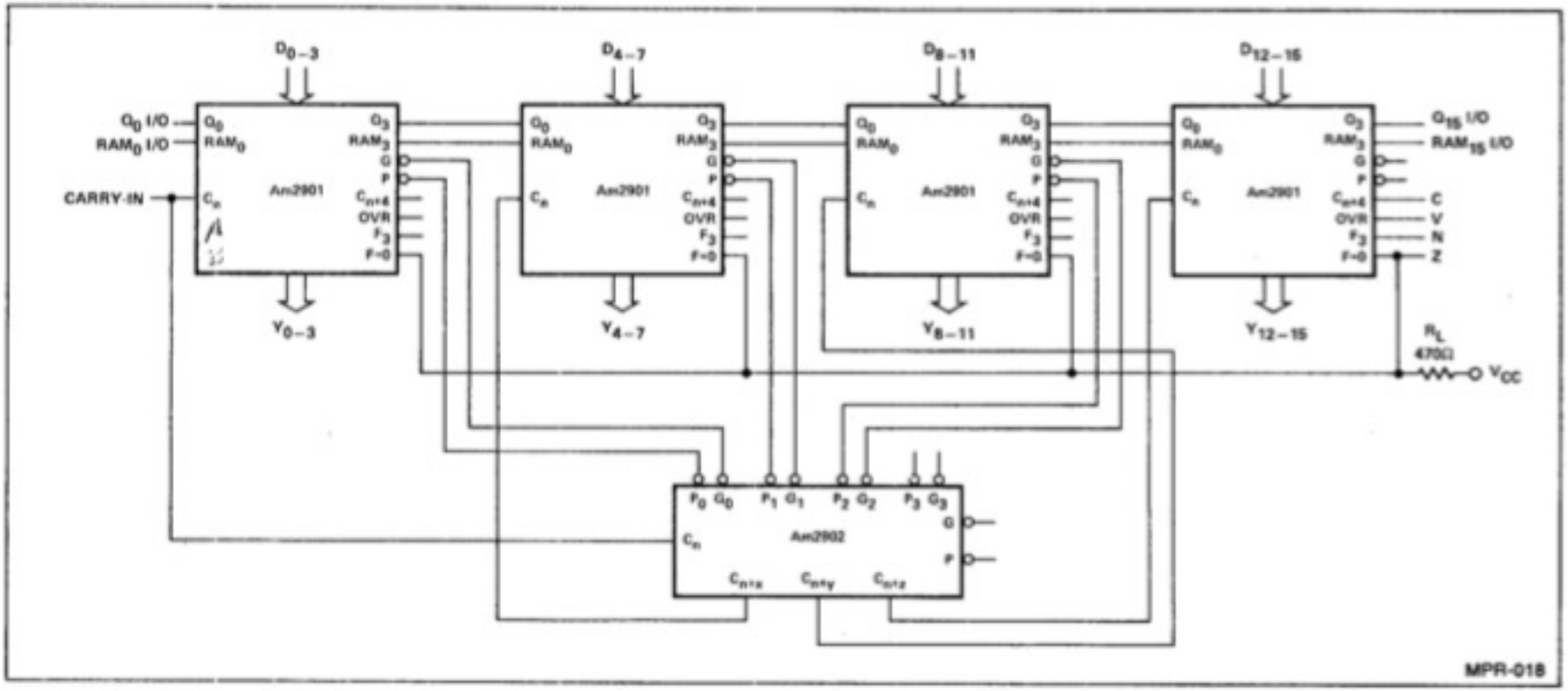
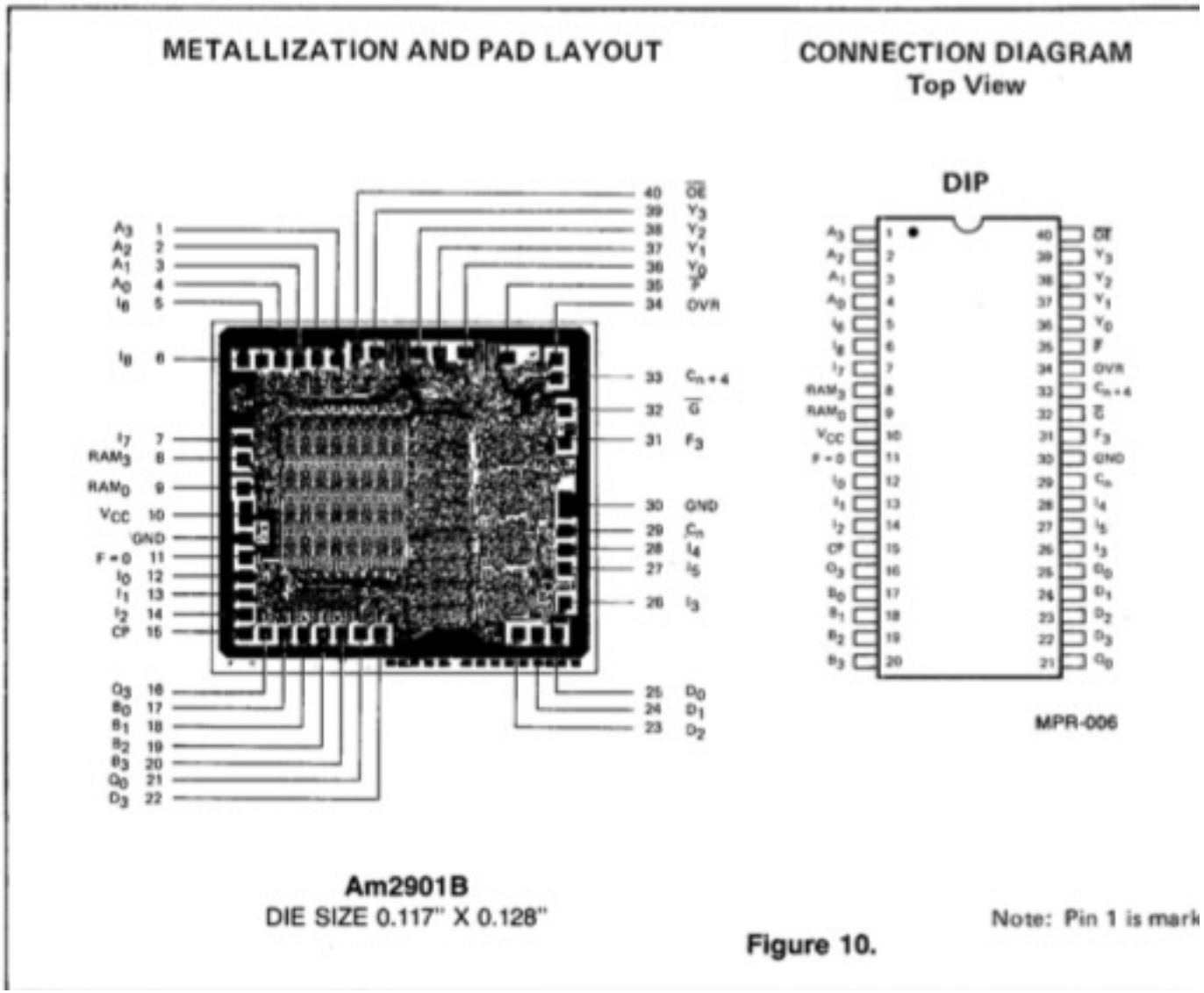


Figure 17. Four Am2901s in a 16-Bit CPU Using the Am2902 for Carry Lookahead.

2901 Chip

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Am2900



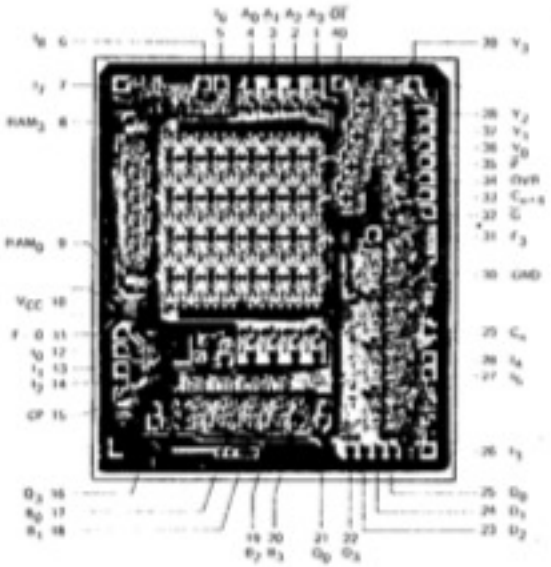
2901-A-B Die

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Am2900

SMALLER DIE SIZES MAKE FASTER PARTS

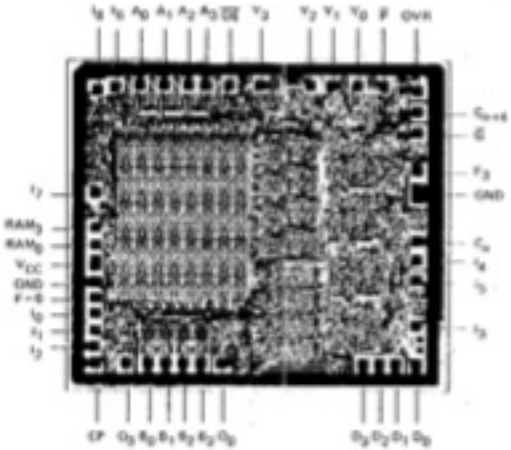
Am2901



DIE SIZE 0.167" X 0.198"

Speed A,B → $\overline{G}$,$\overline{P}$ 80ns

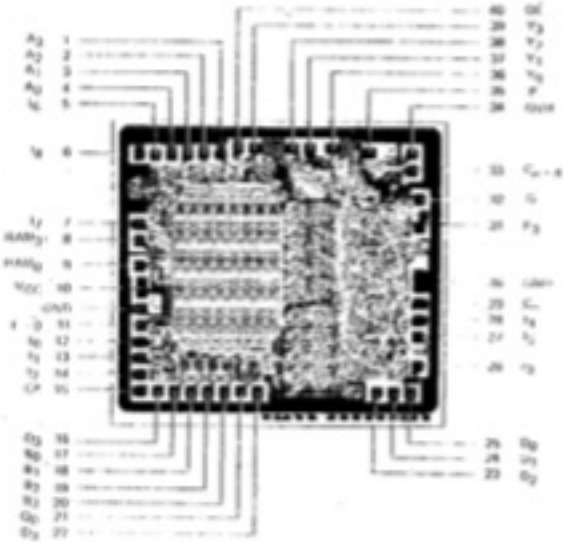
Am2901A



DIE SIZE 0.132" X 0.149"

Speed A,B → $\overline{G}$,$\overline{P}$ 65ns

Am2901B



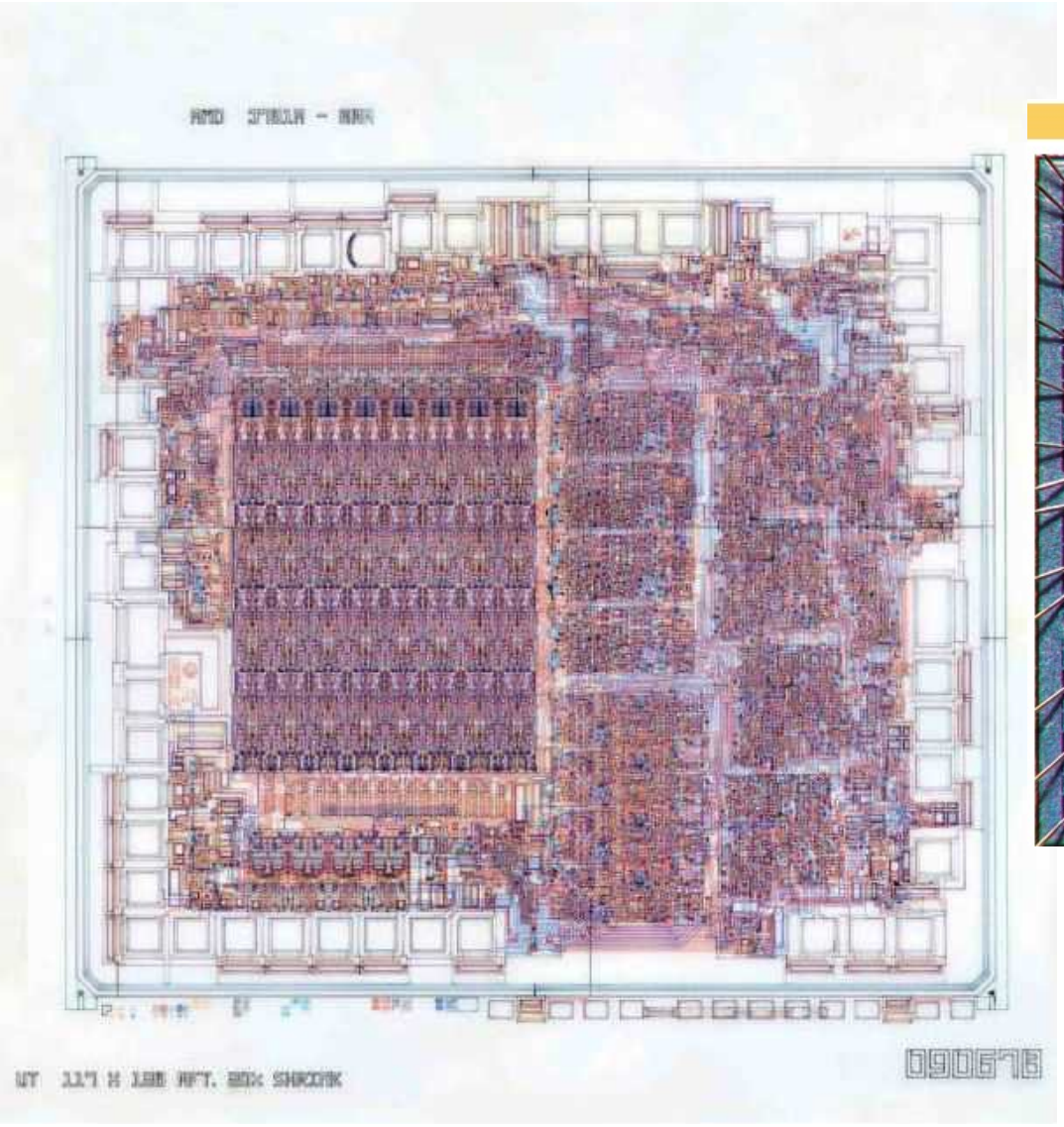
DIE SIZE 0.117" X 0.128"

Speed A,B → $\overline{G}$,$\overline{P}$ 50ns

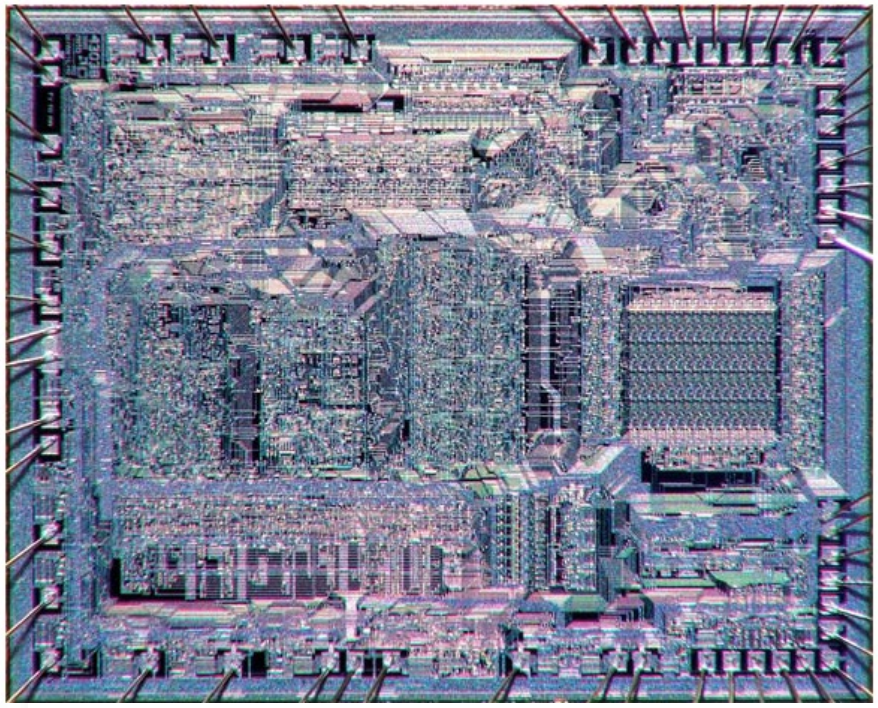
Am2901/3 4-bit MPU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Bit-slice 1975-85



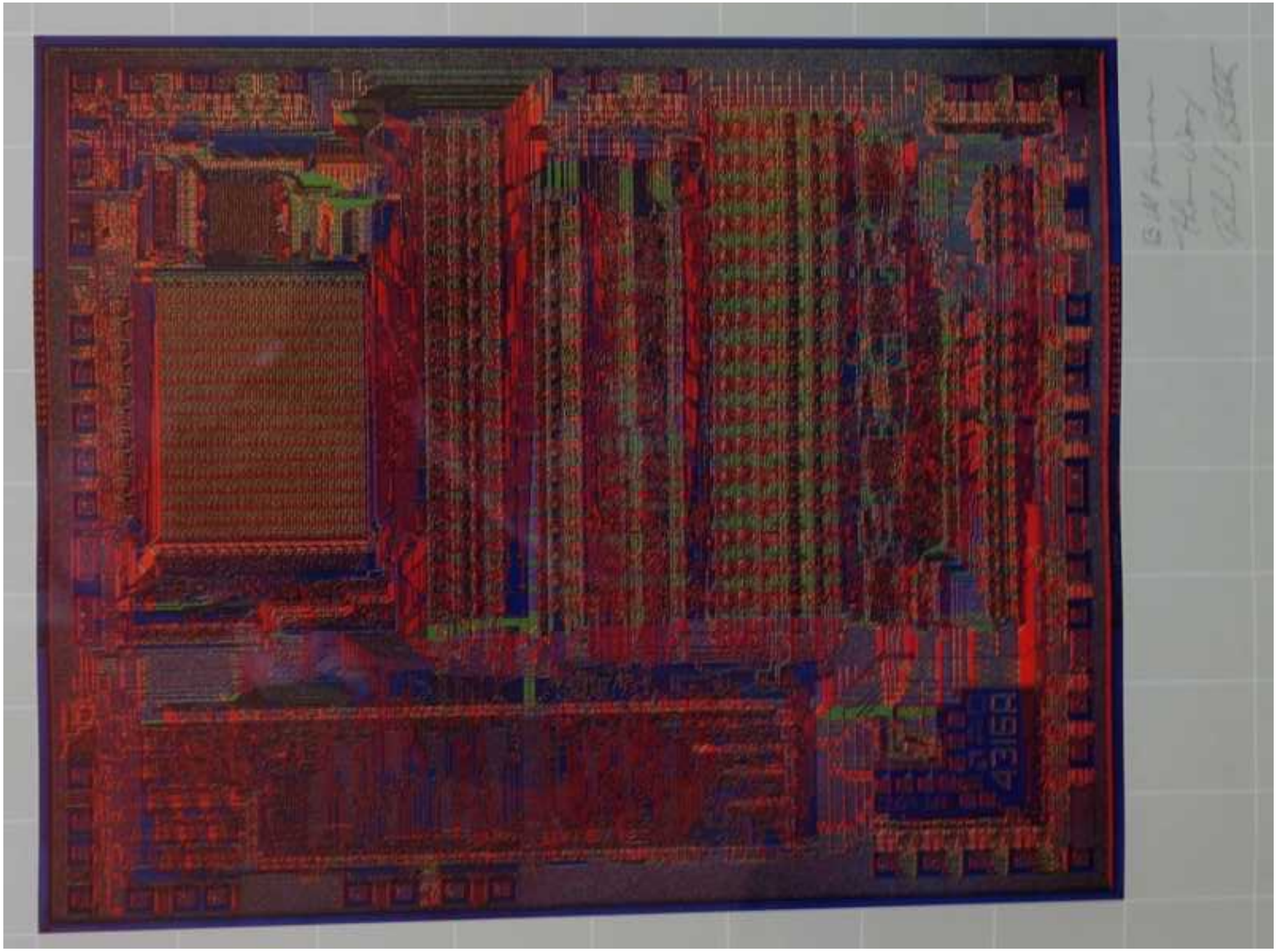
AMD AM2903ADC



Am29116 16-bit MPU

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Bit-slice 1985-88



Section

DR JEFF SOFTWARE
INDIE APP DEVELOPER

Jeff Drobman
©2016-23

Debug & Test

ICE Debugger

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

called setting a breakpoint in the code. The debugger could show the contents of the registers and then we could tell the debugger to jump back into the **loop** function and continue execution of our program.

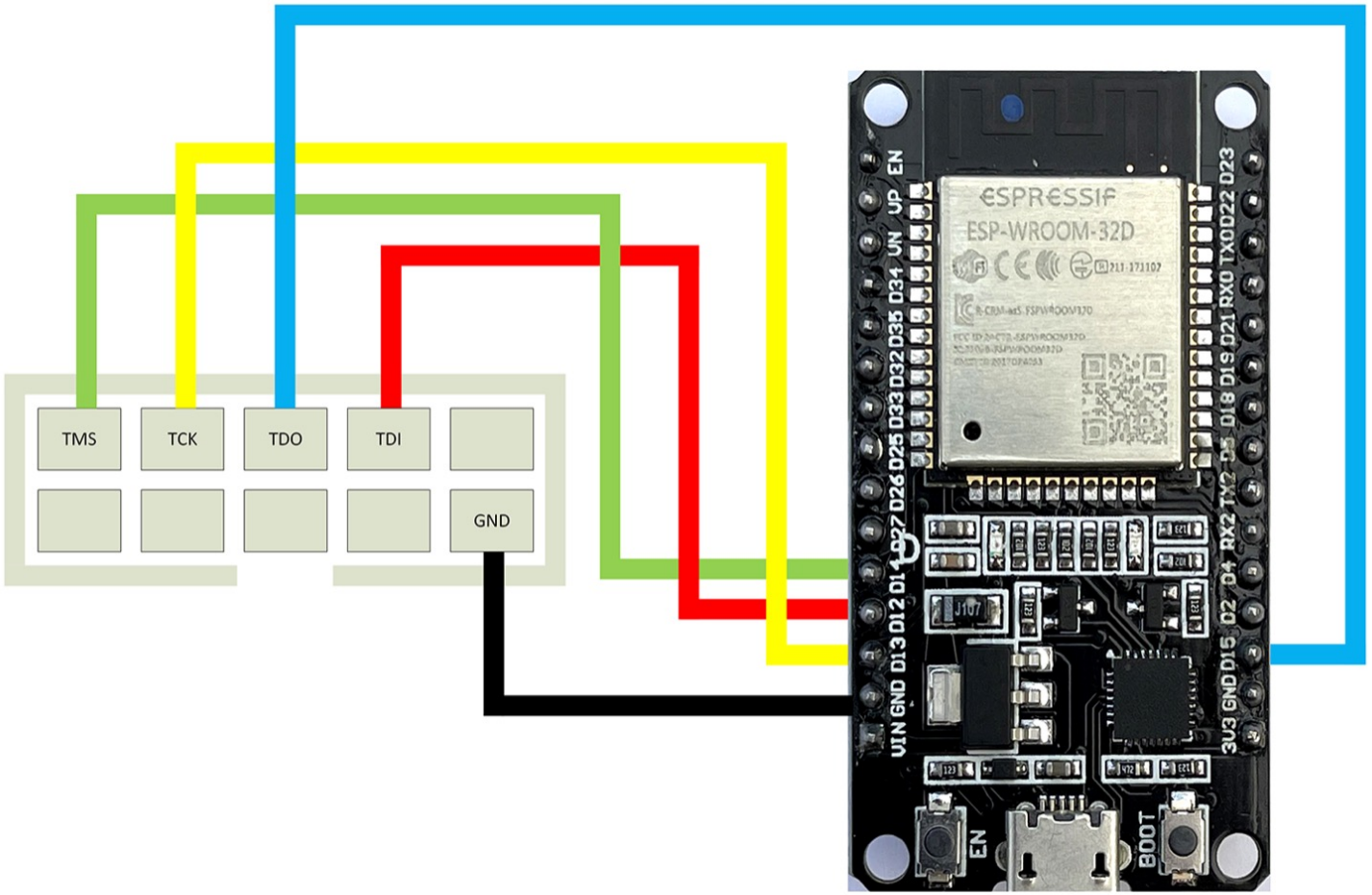


Figure 3: Wiring diagram for connecting an ESP32-PROG (left) to a DOIT ESP32 (right)

ICE Debugger

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

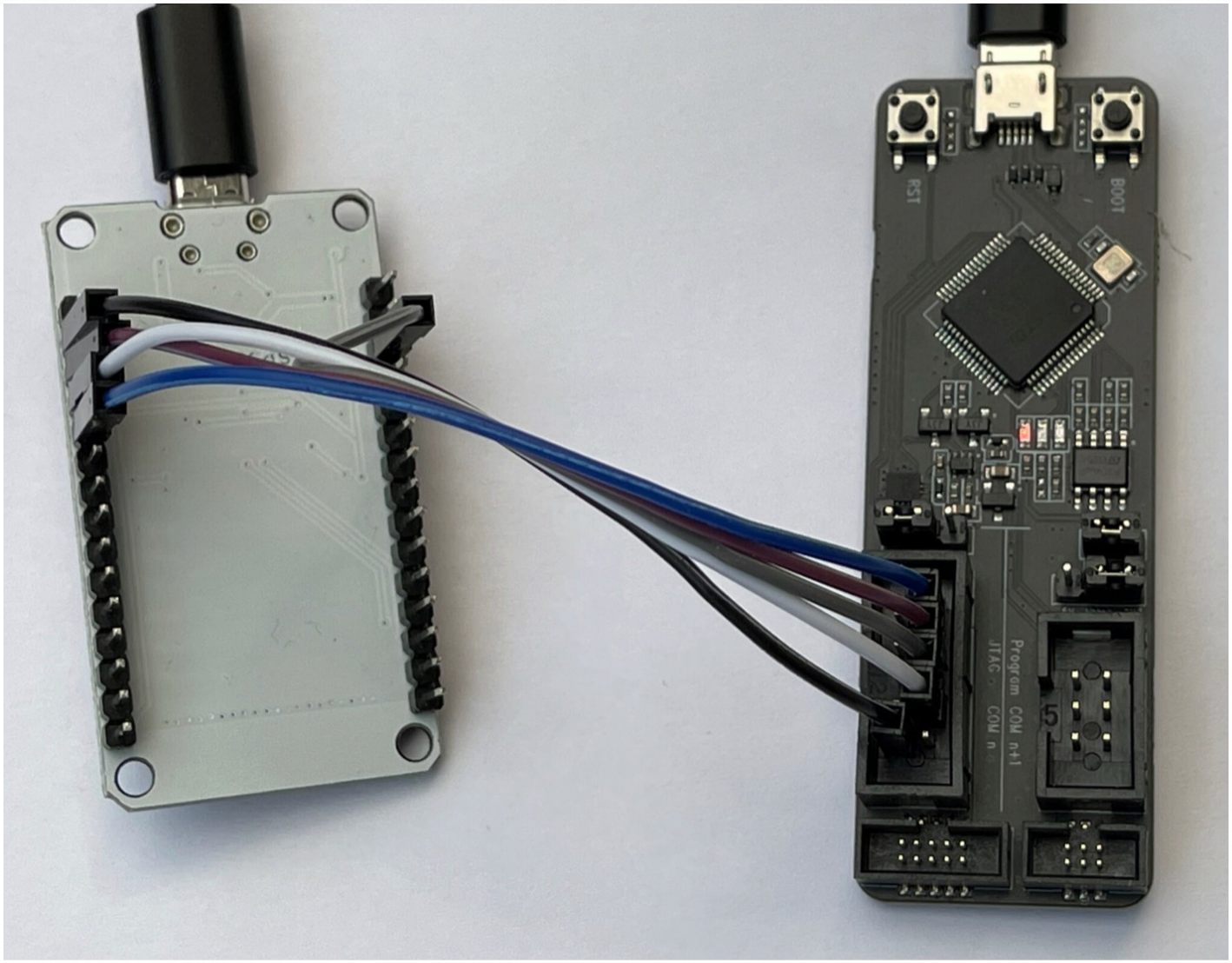


Figure 1: An ESP32-PROG (on the right) connected to a DOIT ESP32 device using direct connection to the JTAG pins

ICE Debugger

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

ARMv7

Address	Machine Code	Opcode	Operands	Description
83:	fd2591	l32r	a9, 40018	Make register a9 point at the location where i is stored
86:	0988	l32i.n	a8, a9, 0	Load the location pointed at by a9 into a8
88:	881b	addi.n	a8, a8, 1	Add 1 to the value in a8
8a:	0989	s32i.n	a8, a9, 0	Store a8 in the location pointed at by a9
8c:	fd2491	l32r	a9, 4001c	Make register a9 point at the location where j is stored
8f:	0988	l32i.n	a8, a9, 0	Load the location pointed at by a9 into a8
91:	880b	addi.n	a8, a8, -1	Subtract 1 from the value in a8
93:	0989	s32i.n	a8, a9, 0	Store a8 in the location pointed at by a9

Figure 2: ESP32 Assembler table

Additional Material

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

JTAG

JEDEC

JTAG

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Enter JTAG

As the power and complexity of microprocessors grew, it became harder to make bond-outs to expose all the internal signals that make hardware debugging possible. To address this, manufacturers formed a Joint Tag Action Group (JTAG) to define standards by which a device can expose its internal state using just a few pins.

Many circuit boards have pins labelled JTAG which are used during manufacture and testing. Sometimes these pins can also be used for hardware debugging. Not all processors support hardware debugging connections. The ATmega328P processor used in the Arduino Uno cannot be debugged in this way. However, the ESP32 does provide these connections. Some of the general-purpose input/output (GPIO) pins on an ESP32 can be used as JTAG connectors. To debug code running in hardware, you'll need some way of connecting your development computer to the JTAG signals on the target device. Espressif (the same company that makes the ESP32) produces a great device for this. It is called the ESP32-PROG.

JTAG

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wikipedia

Boundary scan

From Wikipedia, the free encyclopedia
(Redirected from JTAG boundary scan)

Boundary scan is a method for testing interconnects (wire lines) on printed circuit boards or sub-blocks inside an integrated circuit. Boundary scan is also widely used as a debugging method to watch integrated circuit pin states, measure voltage, or analyze sub-blocks inside an integrated circuit.

The Joint Test Action Group (JTAG) developed a specification for boundary scan testing that was standardized in 1990 as the IEEE Std. 1149.1-1990. In 1994, a supplement that contains a description of the Boundary Scan Description Language (BSDL) was added which describes the boundary-scan logic content of IEEE Std 1149.1 compliant devices. Since then, this standard has been adopted by electronic device companies all over the world. Boundary scan is now mostly synonymous with JTAG.[1][2]

Debugging [edit]

The boundary scan architecture also provides functionality which helps developers and engineers during development stages of an embedded system. A JTAG Test Access Port (TAP) can be turned into a low-speed logic analyzer.

History [edit]

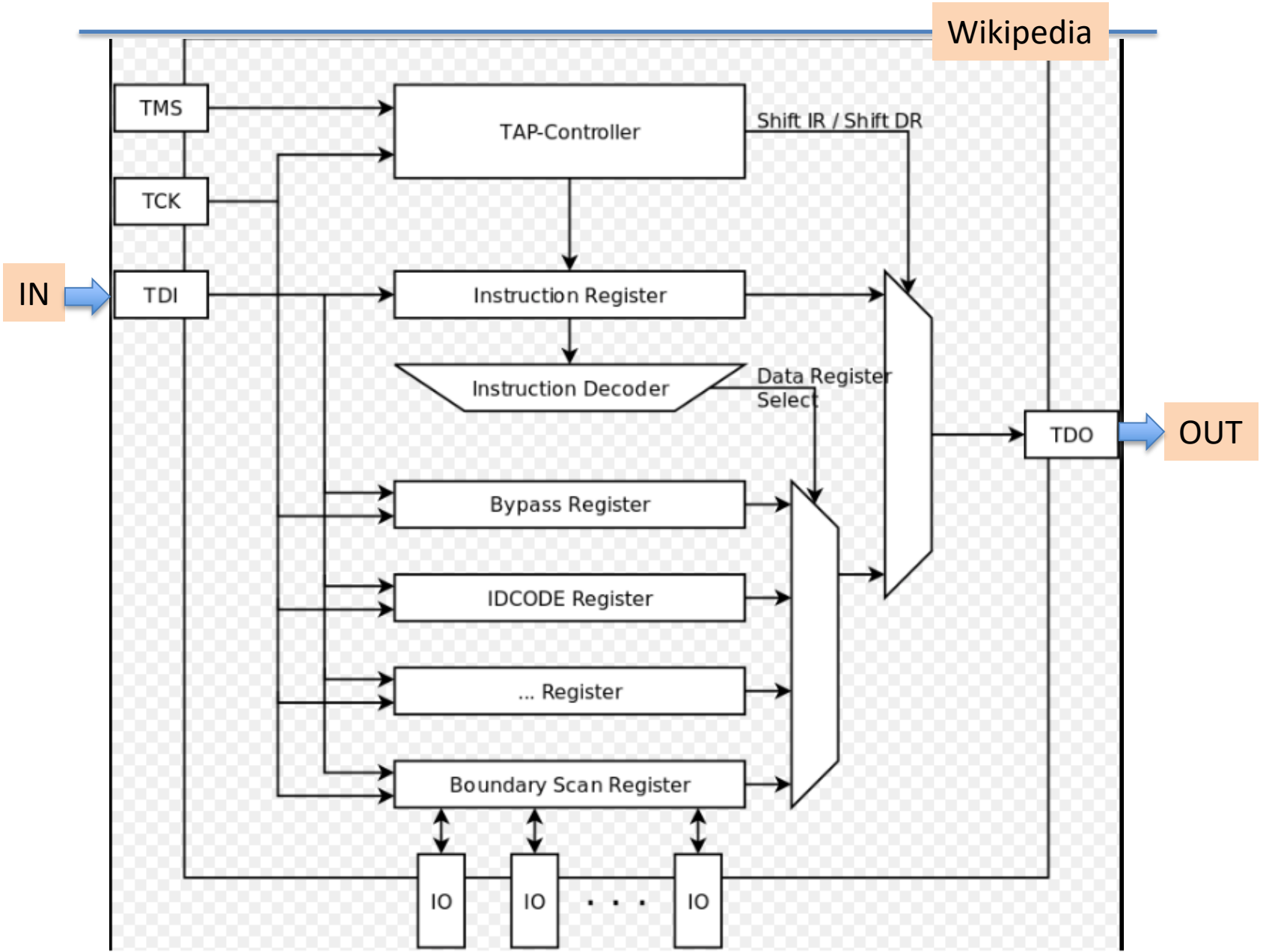
James B. Angell at Stanford University proposed serial testing.[4]

IBM developed level-sensitive scan design (LSSD).[5][6]

JTAG

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wikipedia



JTAG

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wikipedia

On-chip infrastructure [edit]

To provide the boundary scan capability, IC vendors add additional logic to each of their devices, including *scan cells* for each of the external traces. These cells are then connected together to form the external boundary scan shift register (BSR), and combined with JTAG Test Access Port (TAP) controller support comprising four (or sometimes more) additional pins plus control circuitry.

Some TAP controllers support scan chains between on-chip logical design blocks, with JTAG instructions which operate on those internal scan chains instead of the BSR. This can allow those integrated components to be tested as if they were separate chips on a board. On-chip debugging solutions are heavy users of such internal scan chains.

These designs are part of most Verilog or VHDL libraries. Overhead for this additional logic is minimal, and generally is well worth the price to enable efficient testing at the board level.

For normal operation, the added boundary scan latch cells are set so that they have no effect on the circuit, and are therefore effectively invisible. However, when the circuit is set into a test mode, the latches enable a data stream to be shifted from one latch into the next. Once a complete data word has been shifted into the circuit under test, it can be latched into place so it drives external signals. Shifting the word also generally returns the input values from the signals configured as inputs.

Test mechanism [edit]

As the cells can be used to force data into the board, they can set up test conditions. The relevant states can then be fed back into the test system by clocking the data word back so that it can be analyzed.

By adopting this technique, it is possible for a test system to gain test access to a board. As most of today's boards are very densely populated with components and tracks, it is very difficult for test systems to physically access the relevant areas of the board to enable them to test the board. Boundary scan makes access possible without always needing physical probes.

In modern chip and board design, Design For Test is a significant issue, and one common design artifact is a set of boundary scan test vectors, possibly delivered in Serial Vector Format (SVF) or a similar interchange format.

JTAG

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

Wikipedia

JTAG test operations [edit]

Devices communicate to the world via a set of input and output pins. By themselves, these pins provide limited visibility into the workings of the device. However, devices that support boundary scan contain a shift-register cell for each signal pin of the device. These registers are connected in a dedicated path around the device's boundary (hence the name). The path creates a virtual access capability that circumvents the normal inputs and provides direct control of the device and detailed visibility at its outputs.[3] The contents of the boundary scan are usually described by the manufacturer using a part-specific BSDL file.

Among other things, a BSDL file will describe each digital signal exposed through pin or ball (depending on the chip packaging) exposed in the boundary scan, as part of its definition of the Boundary Scan Register (BSR). A description for two balls might look like this:

```
"541 (bc_1,                  *,  control,  1)," &
"542 (bc_1,        GPIO51_ATACS1,  output3,  X,    541,   1,   Z)," &
"543 (bc_1,        GPIO51_ATACS1,    input,  X)," &
"544 (bc_1,                  *,  control,  1)," &
"545 (bc_1,        GPIO50_ATACS0,  output3,  X,    544,   1,   Z)," &
"546 (bc_1,        GPIO50_ATACS0,    input,  X)," &
```

That shows two balls on a mid-size chip (the boundary scan includes about 620 such lines, in a 361-ball BGA package), each of which has three components in the BSR: a control configuring the ball (as input, output, what drive level, pullups, pulldowns, and so on); one type of output signal; and one type of input signal.

There are JTAG instructions to SAMPLE the data in that boundary scan register, or PRELOAD it with values.

During testing, I/O signals enter and leave the chip through the boundary-scan cells. Testing involves a number of test vectors, each of which drives some signals and then verifies that the responses are as expected. The boundary-scan cells can be configured to support external testing for interconnection between chips (EXTEST instruction) or internal testing for logic within the chip (INTEST instruction).

JEDEC (EIA)

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

JEDEC®
Global Standards for the Microelectronics Industry

STANDARDS & DOCUMENTS | COMMITTEES | NEWS | EVENTS & MEETINGS

GRAPHICS DOUBLE DATA RATE 6 (GDDR6) SGRAM STANDARD

JESD250B

Published: Nov 2018

This document defines the Graphics Double Data Rate 6 (GDDR6) Synchronous Graphics Random Access Memory (SGRAM) specification, including features, functionality, package, and pin assignments. The purpose of this Specification is to define the minimum set of requirements for 8 Gb through 16 Gb x16 dual channel GDDR6 SGRAM devices. System designs based on the required aspects of this standard will be supported by all GDDR6 SGRAM vendors providing compatible devices. Some aspects of the GDDR6 standard such as AC timings and capacitance values were not standardized. Some features are optional and therefore may vary among vendors. In all cases, vendor data sheets should be consulted for specifics. This document was created based on some aspects of the GDDR5 Standard (JESD212). Item 1836.99D.

JEDEC (EIA)

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23

JEDEC History

In 1924, the Radio Manufacturers Association (which later became the Electronic Industries Association) was established. In 1944, the Radio Manufacturers Association and the National Electronic Manufacturers Association established the Joint Electron Tube Engineering Council (JETEC), which was responsible for assigning and coordinating type numbers of electron tubes. As the radio industry expanded into the emerging field of electronics, various divisions of the EIA, including JETEC, began to function as semi-independent membership groups. The Council expanded its scope to include solid state devices, and by 1958 the organization was renamed the Joint Electron Device Engineering Council (JEDEC) – one council for tubes and one for semiconductors.

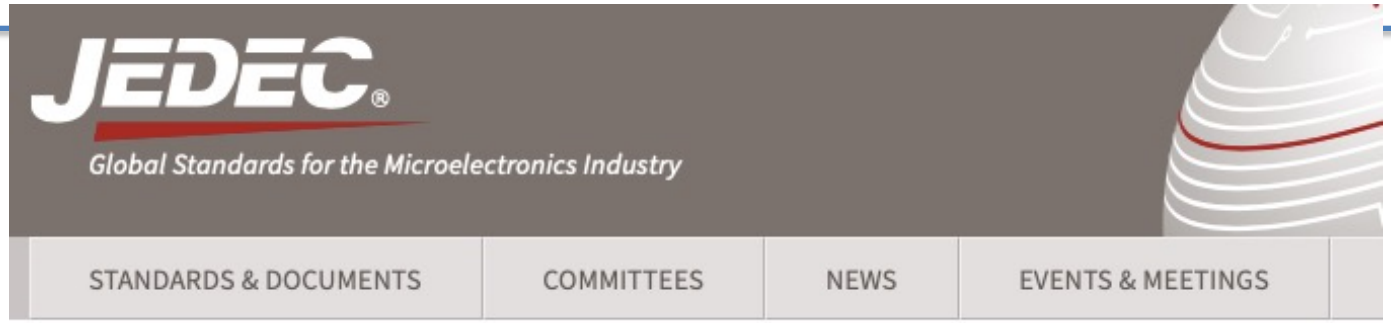
Timeline

- Pre-1960s
- 1960s
- 1970s
- 1980s
- 1990s
- 2000s
- 2010s

JEDEC initially functioned within the engineering department of EIA where its primary activity was to develop and assign part numbers to devices. Over the next 50 years, JEDEC's work expanded into developing test methods and product standards that proved vital to the development of the semiconductor industry. Among the landmark standards that have come from JEDEC committees are:

JEDEC (EIA)

DR JEFF SOFTWARE
INDIE APP DEVELOPER
Jeff Drobman
©2016-23



Why JEDEC Standards Matter

JEDEC committees develop open standards, which are the basic building blocks of the digital economy and form the bedrock on which healthy, high-volume markets are built. For example, JEDEC semiconductor memory standards - from dynamic RAM chips and memory modules to DDR synchronous DRAM and flash components – have enabled huge markets in PCs, servers, digital cameras, MP3 players, smart phones, automotive and HDTV, to name just a few.

Standards enable innovation, serving to commoditize components by lowering their prices while maintaining quality and reliability. This leads suppliers to compete more vigorously on innovative features and gives buyers more variety and a broader selection. The end result is a much larger market than proprietary products could foster, which means more potential sales and revenue.

Standards allow companies to invest more strategically in R&D rather than inventing everything from scratch. Once common form factors are set, companies can base their designs on standards and focus on innovation.