

CMSC 411 – Section 03

Computer Architecture

Dr. Ergün Simşek

Assistant Professor of CSEE

Director of Graduate Data Science Programs

Fall 22 - Lecture 1

Credits

- Some of these slides were developed by Profs. Mohamed Younis, Jeannette Kartchner, David Black-Schaffer, Onur Mutlu, Don Porter, Leonard McMillan, Gary Bishop, Alok Choudhary, and Montek Singh.
- Some slides and other materials are from the book publisher.

People

Instructor: **Dr. Ergun Simsek**

E-mail: simsek@umbc.edu

Office: <https://umbc.webex.com/meet/simsek>

Office hours: Wednesdays 10:00 - 12:00 noon

Research: Photonics, scientific computing, and machine learning.

TA: **Saquib Ahmed**

Office: TBA

E-mail: NQ41459@umbc.edu

Office hours: Monday – Wednesday 10:30 am – 12:30 noon

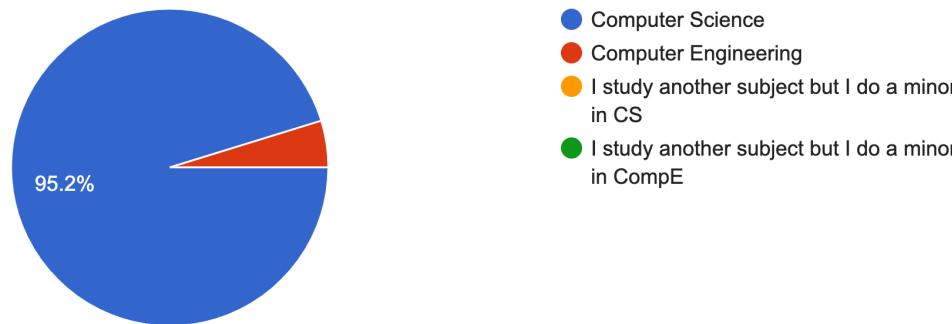
Grader: **Shalini Nomula**

E-mail: OM77059@umbc.edu

You!

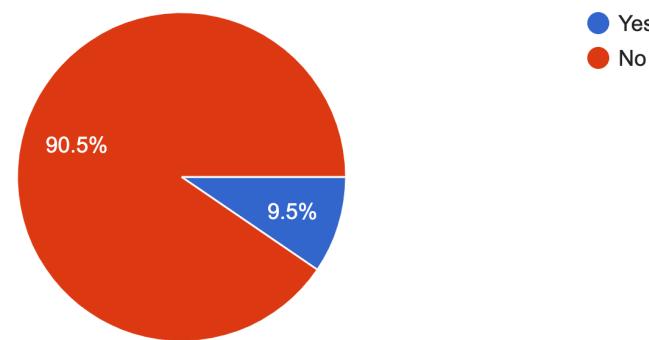
What do you study at UMBC?

21 responses



Have you ever used a Zybook before?

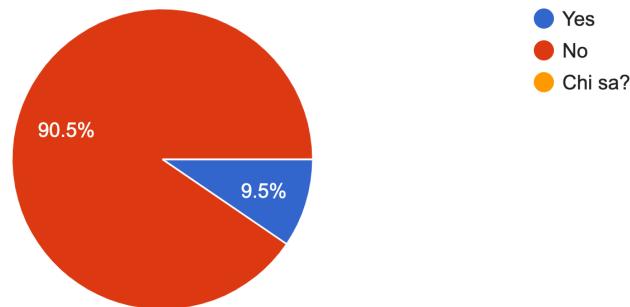
21 responses



You!

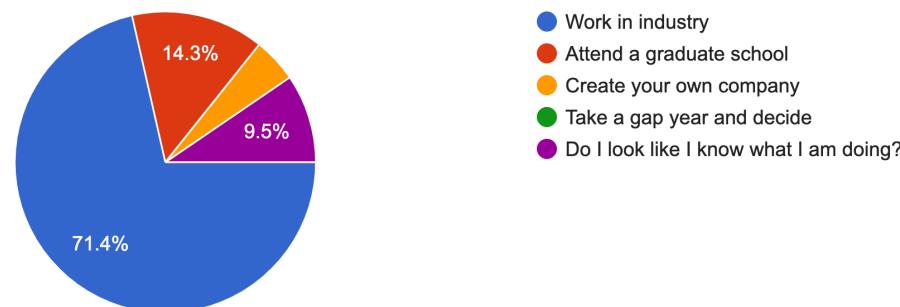
Are you going to graduate at the end of Fall 22?

21 responses



What is your plan after graduation?

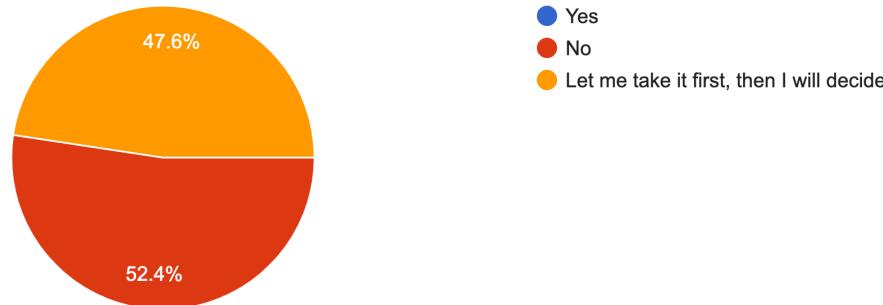
21 responses



You!

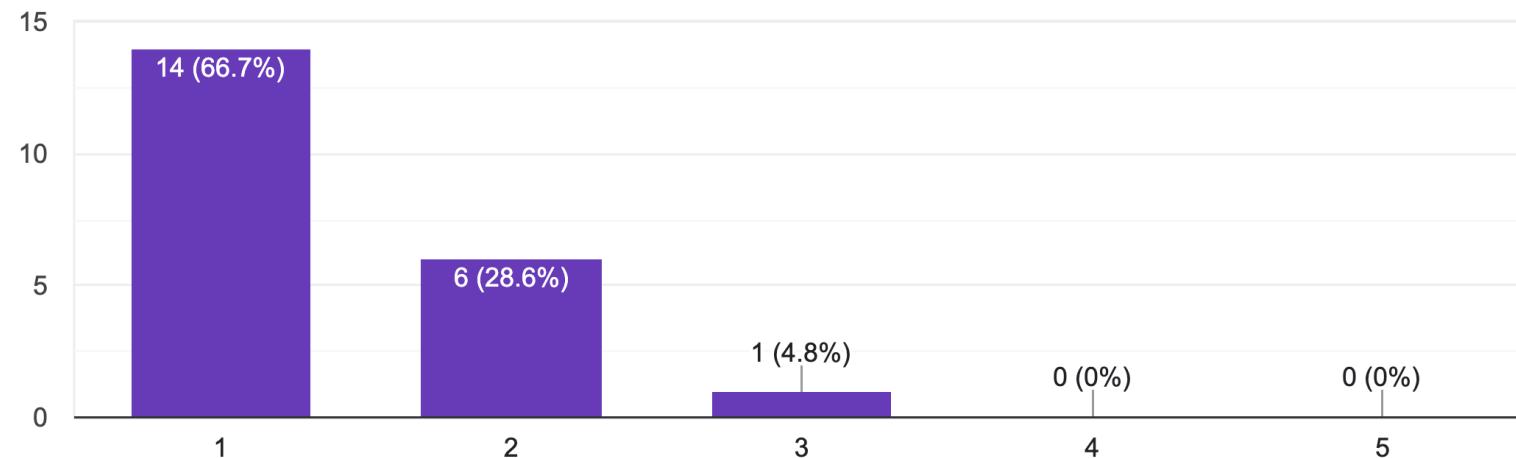
Are you planning to work/conduct research in the field of computer architecture after graduation?

21 responses



Are you familiar with MIPS Assembly language?

21 responses

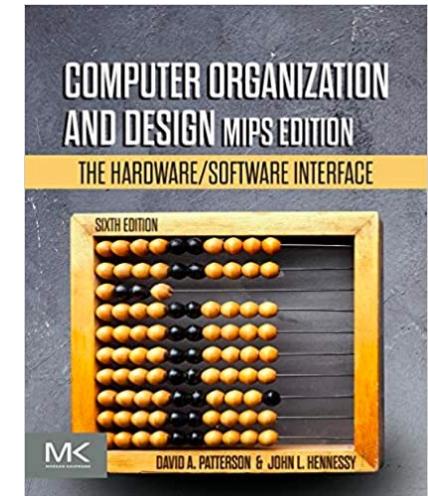


Your Answers to What do you expect to learn in CMSC 411?

- To gain a better understanding on the organization and implementation of varies architectural styles, such as parallel and multi-cycle processing
- I expect to learn MIPS assembly and to see where it can be useful to use ARM vs MIPS.
- Hopefully tie together everything else I learned in cmsc classes - coding, data structures - in practical ways.
- I want to learn how computers are made internally and how they are designed. Maybe as a self project I can make one from scratch.
- How computer architecture works, to have a good idea of what the field has to offer to decide if I want to specialize in it.
- Basic computer architecture knowledge so I am better able to design and structure games I am making based on limitations

Textbook

Computer Organization and Design:
The Hardware/Software Interface
by Patterson and Hennessy



DO NOT PURCHASE A HARDCOPY

\$72

We'll use ZYBOOKS

1. Sign in or create an account at <https://learn.zybooks.com/signup>
Important Note: For signing up, you need to use your UMBC email account in the following format: CampusID@umbc.edu
2. Enter zyBook code: **UMBCCMSC411SimsekFall2022**
3. Subscribe

Tentative Syllabus

1. Performance Evaluation

- Measures of performance
- Benchmarks and metrics

2. Instruction Set Architecture

- Instruction formats & semantics
- Addressing modes **(3)**
- Procedures and Stacks

3. Machine Arithmetic

- ALU design
- Integer multiplication & division
- Floating-point arithmetic

4. Processor Design

- Datapath design
- Instruction exec. & sequencing
- Hardwired & microcode control

5. Performance boosting features

- Pipelining **(3)**
- Instruction level parallelism

6. Memory Hierarchy

- Cache design & evaluation
- Virtual addressing
- Performance evaluation

7. Input/Output

- Types of I/O devices
- Device access and interface
- Device control
- I/O performance

8. Multiprocessor (time permitting)

- Interconnection networks
- Programming issues

Tentative Course Load

Homework Assignments

- ❑ 6-7 assignments will be given and normalized to **%20** of the final grade
- ❑ An average assignments requires about 2-3 hours to perform
- ❑ Assignments are due in class on the due date (not later)

Exams

- ❑ A midterm exam is scheduled on October 24 during scheduled class time (**30%**)
- ❑ The final exam is scheduled on December 16, 1 – 3 pm (**40 %**)
- ❑ Final Exam is comprehensive, but the focus will be on subjects covered after the midterm

Quizzes

- ❑ We will have 4 quizzes (**10%** of your final grade)

Grade Structure and Policy

	Grade distribution	Course grade	Range
Final Exam	40%	A	90% - 100%
Mid-term Exam	30%	B	80% - 89.9%
Homework	20%	C	70% - 79.9%
Quiz	10%	D	60% - 69.9%

- You must do your own work and not copy from anyone else
- Copying/cheating will result in a minimum punishment of a zero grade for the assignment or quiz or the exam

Course Mechanics

□ Policies:

- Problem Sets:
 - You will typically have 1 week to do them.
- Honor Code:
 - The honor code is in effect for all homework, quizzes, exams etc.
Please review the policy on the university website.
- Lecture Notes:
 - I will attempt to make Lecture Slides, Problem Sets, and other course materials available on Blackboard either before class, or soon after, on the day they are given.

Teaching Style and Philosophy

Instructor's role

- Facilitate and guide the students to the fundamental concepts
- Make it simple and elaborate with examples
- Relate as much as possible to available products
- Prepare class notes to be as rich and comprehensive as possible

Exams will question the level of understanding of fundamental concepts

Student's role

- Focus on understanding and digesting the concept
- Do not worry about the grade more than concepts, soon will be a professional
- Slow down the instructor if you do not understand and raise questions
- Participate

TA's role

- Help students with questions related to their assignments
- Resolve computer and tool issues related to the project

Grader's role

- Grade assignments and projects

Engagement!



Mark Sample ✨ @samplereality · Aug 28

...

I no longer call class participation "participation." I call it engagement and emphasize that it's not just about (and in fact may be the opposite of) being talkative in class.

Engagement comes in many forms, not just attendance. Taken holistically, engagement includes (but is not limited to) the following:

Preparation (reviewing readings and material before class)

Focus (avoiding distractions during in-person and online activities)

Presence (Engaged and responsive during group activities)

Asking questions (in class, out of class, online, offline)

Listening (hearing what others say, and also what they're not saying)

Specificity (referring to specific ideas from readings and discussions)

Synthesizing (making connections between readings and discussions)

Prerequisites

- You must have completed CMSC 313 or CMPE 212 and CMPE 310 with a grade of C or better
- You need to know at least the following concepts:
 - basic data types: integers, characters, Boolean, etc.
 - basic arithmetic operators and expressions
 - “if-then-else” constructs, and “while”/“for” loops
 - function and procedure calls
 - basic Boolean operators (AND, OR, XOR, etc.)
- If you don't know many of the above concepts, you might be in the wrong class(room)

How NOT to do well in this course

□ BIG mistakes

- Skipping lectures
- Not reading the book (only reviewing lecture slides)
- Not spending enough time to do homework
 - Start early. Many problem sets are too hard to attempt the night before.
- Not asking questions in class
- Not discussing concepts with other students
 - But all work handed in must be your own (see Honor Code)
- Looking up solutions from earlier semesters = cheating.
Not worth it.

Helpful Links

MIPS Simulators:

- Download SPIM: <http://spimsimulator.sourceforge.net/>
- Download MARS:
<http://courses.missouristate.edu/kenvollmar/mars/download.htm>
- Online manual for MIPS using QtSpim:
<http://www.egr.unlv.edu/~ed/MIPStextSMv11.pdf>

MIPS Reference:

- [Quick Reference](#)
- [MIPS Opcode LookUp](#)
- ["Cheat Sheet" MIPS Reference \(Green Card from the text for use on exams\)](#)

Calculators:

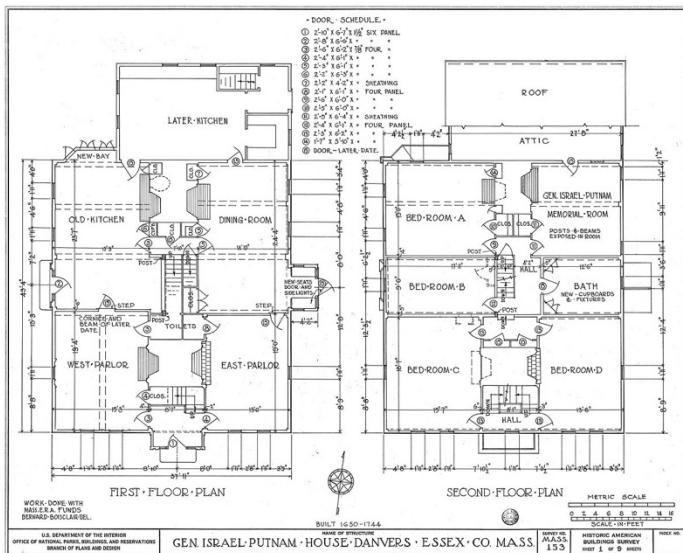
- [Decimal-Binary-Hex Converter](#)
- [IEEE754 Floating Point Calculator](#)
- [Desmos Graphing Calculator](#)

Logic Circuit Simulator
<https://circuitverse.org>
Warning: 32-bit

An Introduction to Computer Architecture and Why Should We Study It?

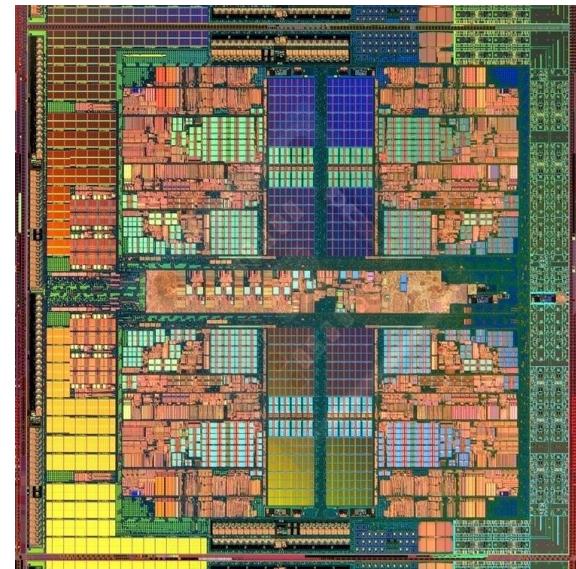
What is Computer Architecture?

Architecture



A House in Barcelona

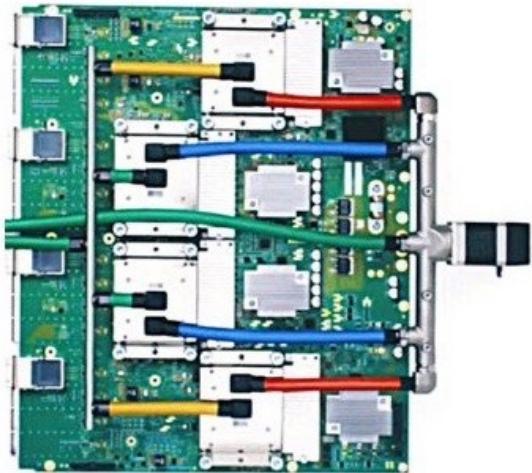
Computer Architecture



AMD Barcelona

The science and art of designing, selecting, and interconnecting hardware components and designing the hardware/software interface to create a computing system that meets **functional**, **performance**, **energy consumption**, **cost**, and other specific goals.

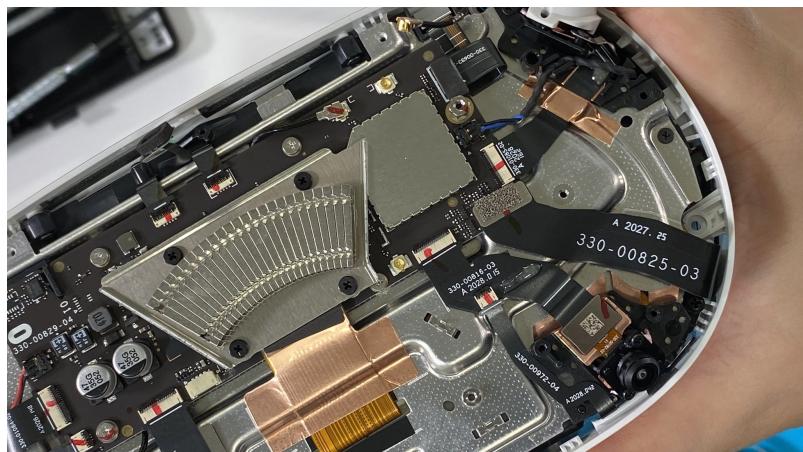
Where do you find computers today?



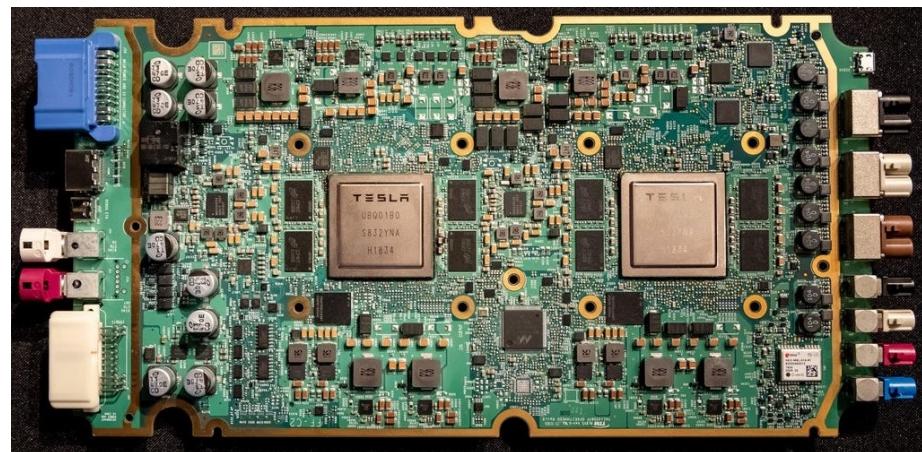
Google – TPU v4



Apple - A15



Oculus – Quest 2



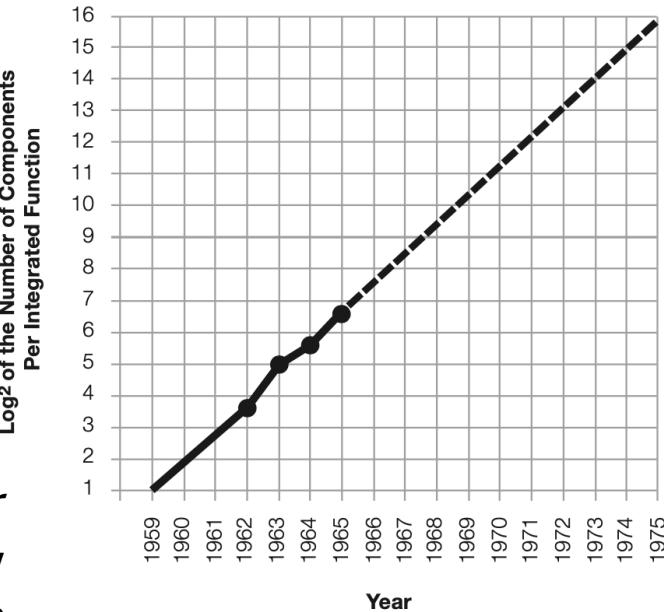
Tesla – FSD Chip

Q: What has fueled progress in computer technology?

A Man Who Can See The Future

Gordon E. Moore, “[Cramming more components onto integrated circuits](#),” Electronics Magazine, 1965.

With unit cost falling as the number of components per circuit rises, by 1975 economics may dictate squeezing as many as 65,000 components on a single silicon chip



Moore's Law: The Number of Transistors

Moore's Law: The number of transistors on microchips doubles every two years

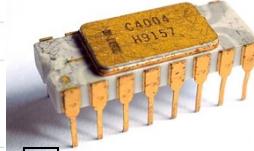
Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important for other aspects of technological progress in computing – such as processing speed or the price of computers.

Our World
in Data

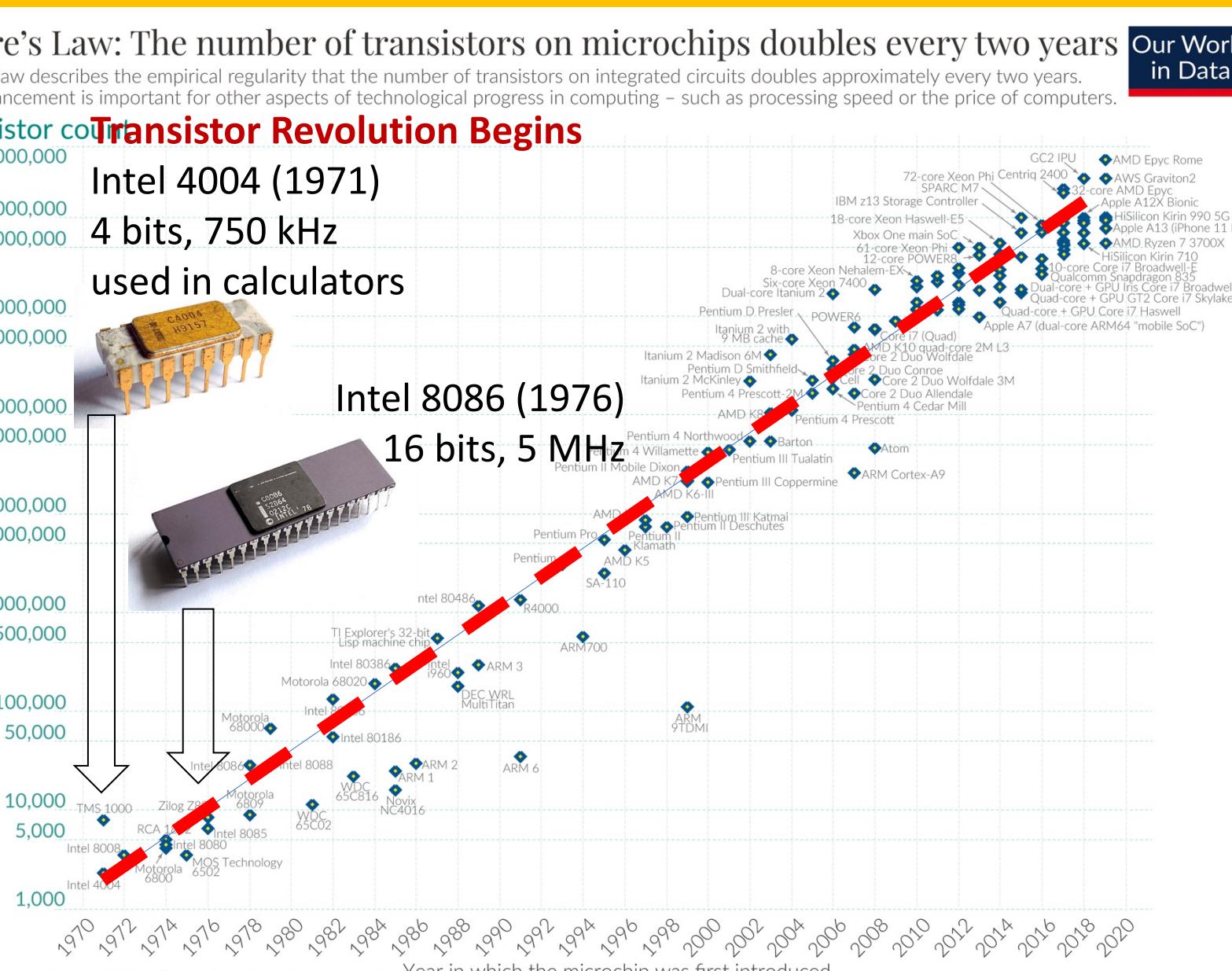
Transistor Revolution Begins

50,000,000,000
10,000,000,000
5,000,000,000
1,000,000,000
500,000,000
100,000,000
50,000,000
10,000,000
5,000,000
1,000,000
500,000
100,000
50,000
10,000
5,000
1,000

Intel 4004 (1971)
4 bits, 750 kHz
used in calculators



Intel 8086 (1976)
16 bits, 5 MHz



Data source: Wikipedia ([wikipedia.org/wiki/Transistor_count](https://en.wikipedia.org/wiki/Transistor_count))

Year in which the microchip was first introduced

OurWorldInData.org – Research and data to make progress against the world's largest problems.

Licensed under CC-BY by the authors Hannah Ritchie and Max Roser.

Revolution II: Implicit Parallelism

Pinnacle of Single-Core Microprocessors

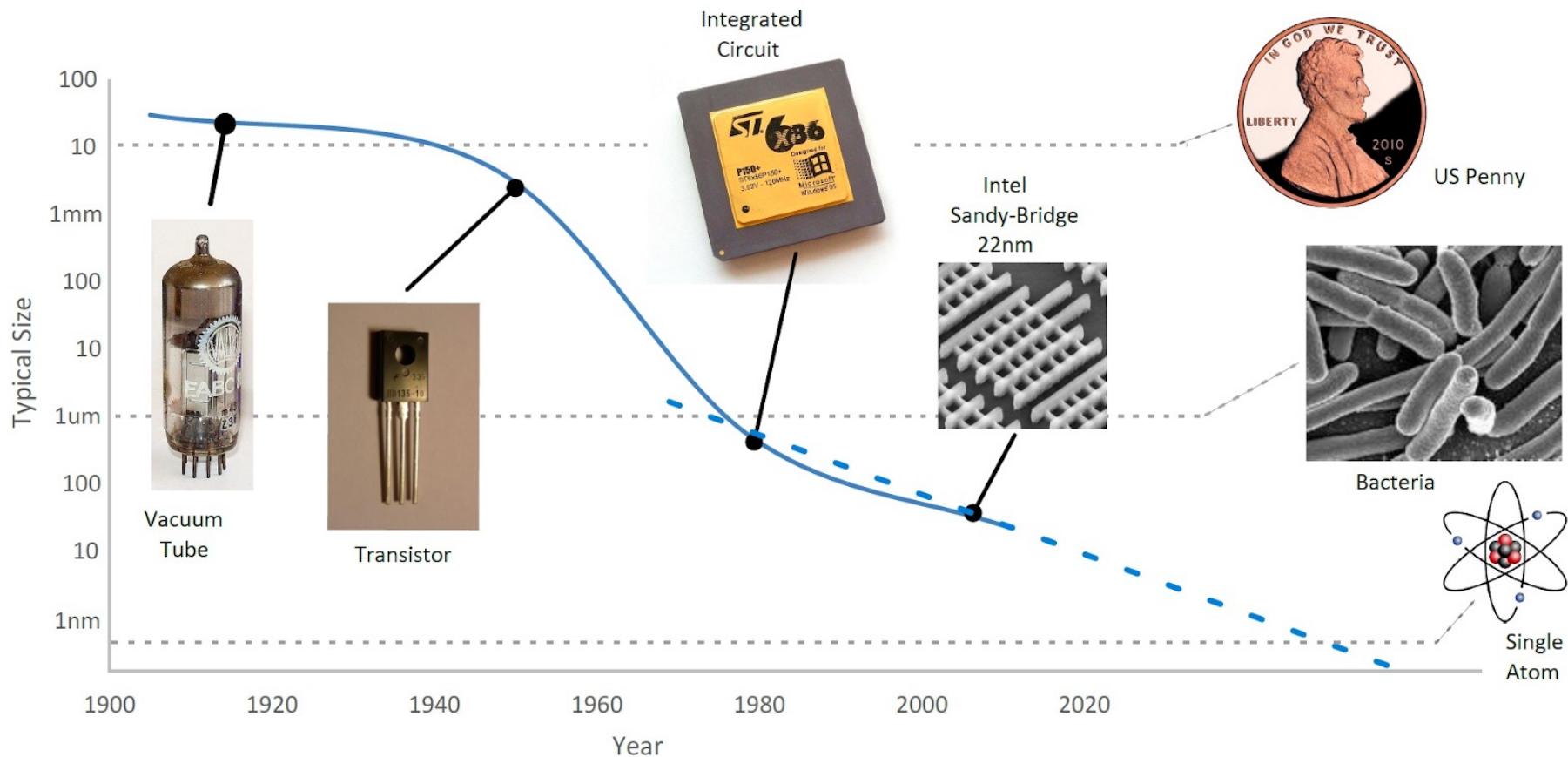
□ Intel Pentium4 (2003)

- Application: desktop/server
- Technology: 90nm (1/100x)
- 55M transistors (20,000x)
- 101 mm² (10x)
- 3.4 GHz (10,000x)
- 1.2 Volts (1/10x)
- 32/64-bit data (16x)
- 22-stage pipelined datapath
- 3 instructions per cycle (superscalar)
- Two levels of on-chip cache
- data-parallel vector (SIMD) instructions, hyperthreading

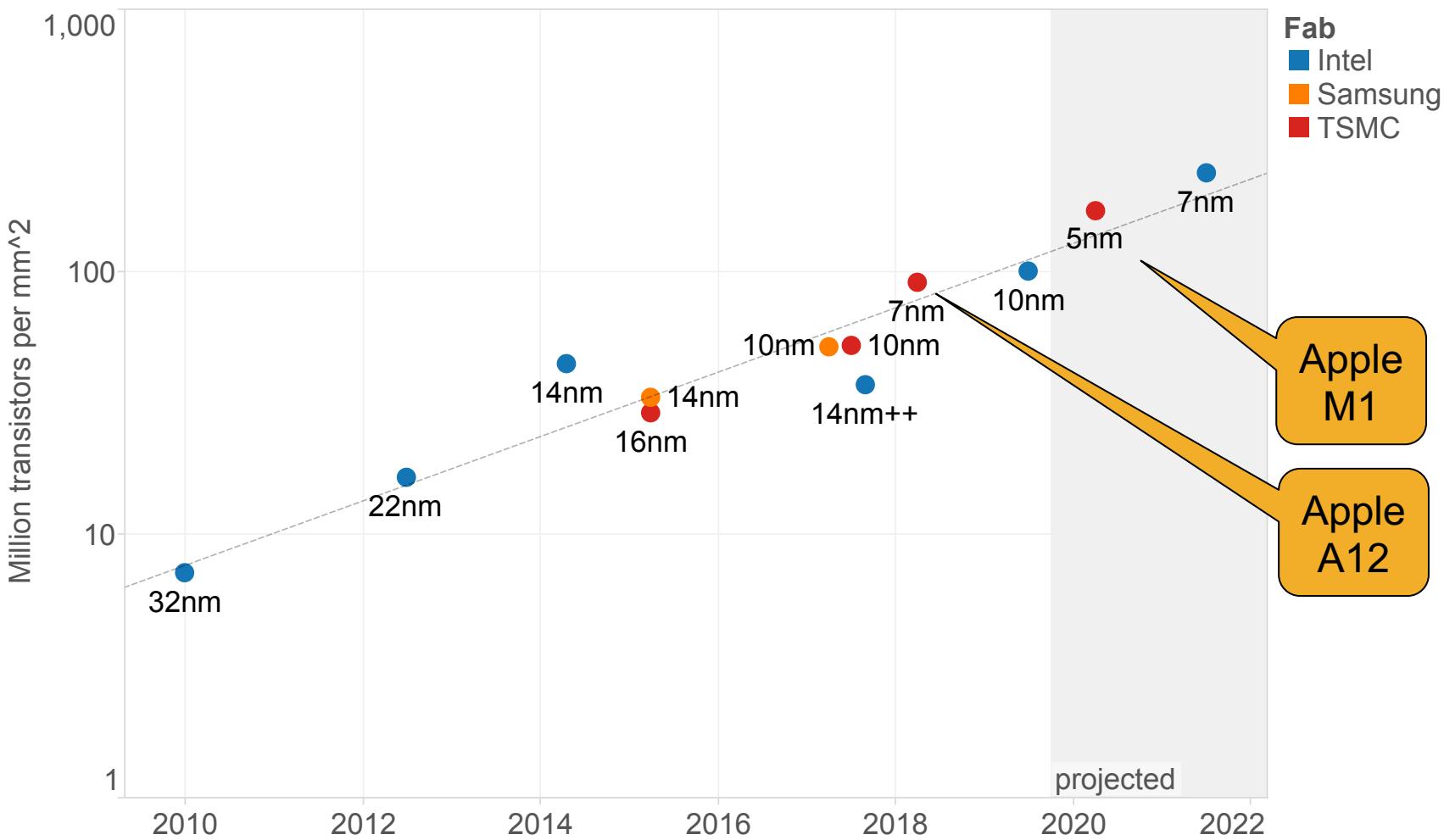


The End of Moore's Law? Physical Limits

Transistor Size over Time



Moore's Law today



Apple M1 MAX

ProRes

encode and decode



Thunderbolt 4



Secure Enclave



Support for four external displays

Up to

64GB

Unified memory

57 billion
Transistors

16-core

**Neural
Engine**

11 trillion operations per second

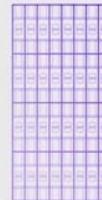


Industry-leading
performance per watt

5 nm process

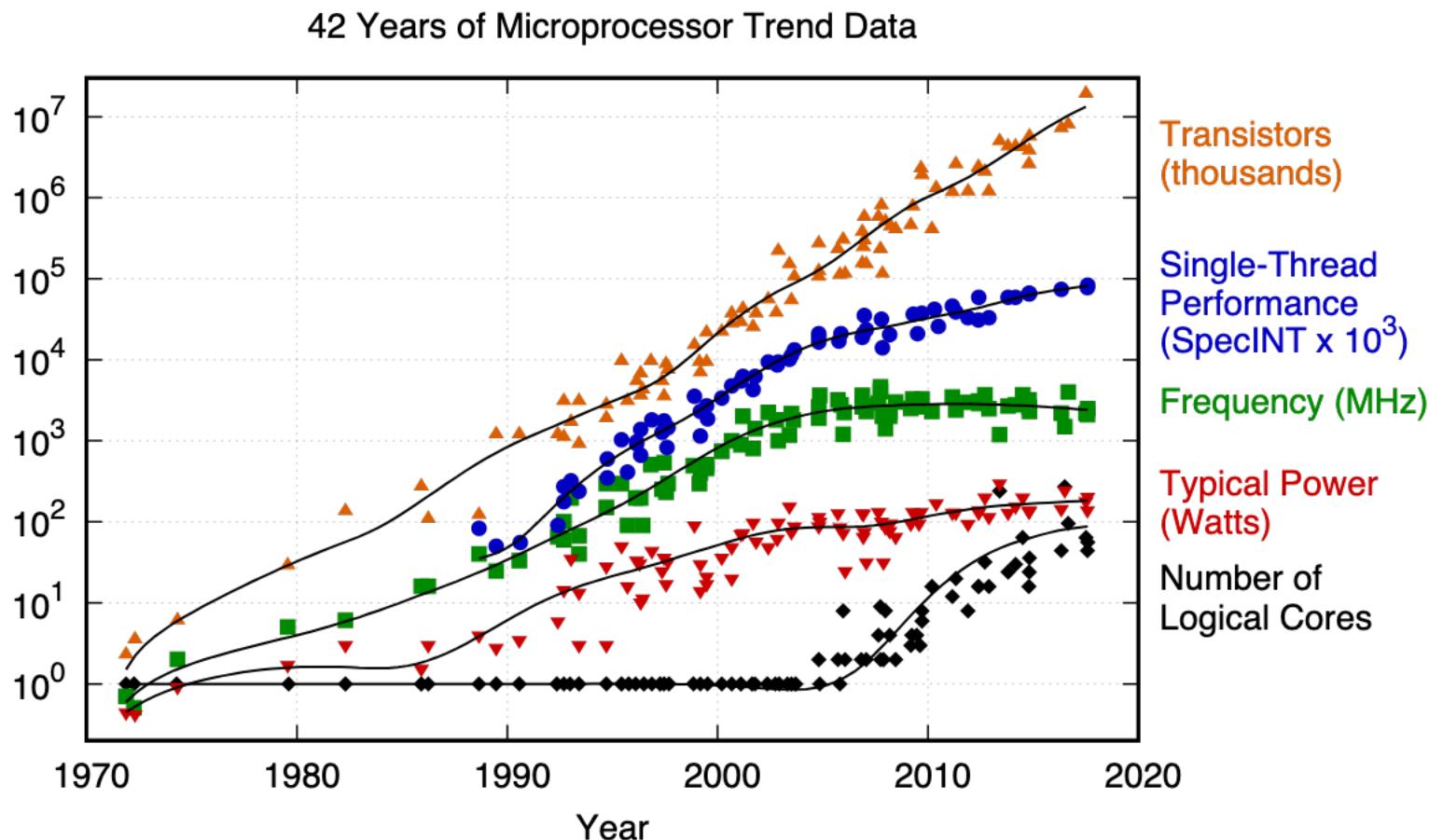
10-core
CPU

Up to
32-core
GPU



400GB/s
Memory bandwidth

The End of Moore's Law? Power Wall

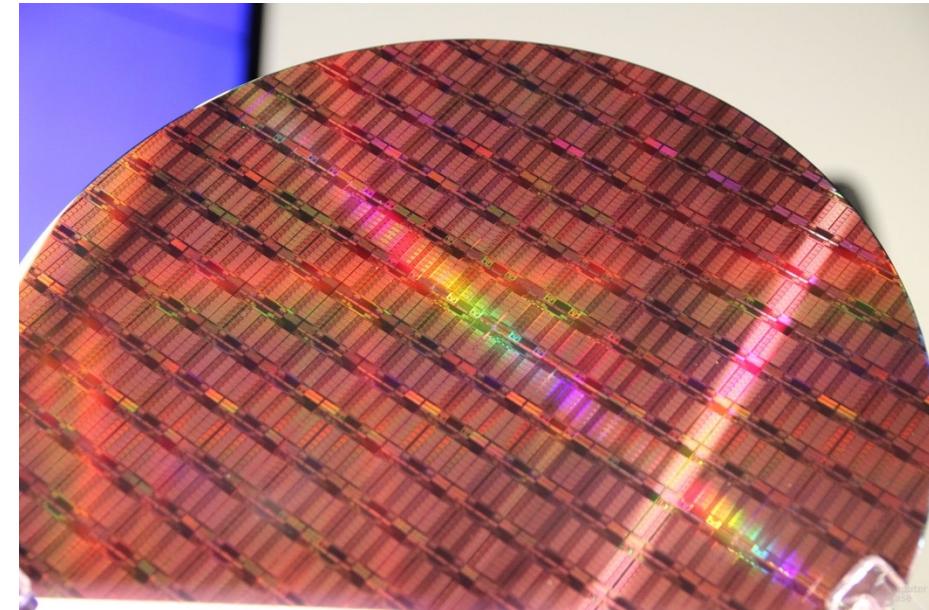


Revolution III: Explicit Parallelism

Modern Multicore Processor

❑ Intel Xeon E5-2699 V4 (2016)

- Application: server
- Technology: 14nm (16% of P4)
- 7.2B transistors (130x)
- 456 mm² (4.5x)
- 2.4 to 3.6 Ghz (~1x)
- 1.8 Volts (~1x)
- 256-bit data (2x)
- 14-stage pipelined datapath (0.5x)
- 4 instructions per cycle (1x)
- Three levels of on-chip cache
- data-parallel vector (SIMD) instructions, hyperthreading
- **22-core multicore** (22x)



Revolution IV: Heterogeneous Processing

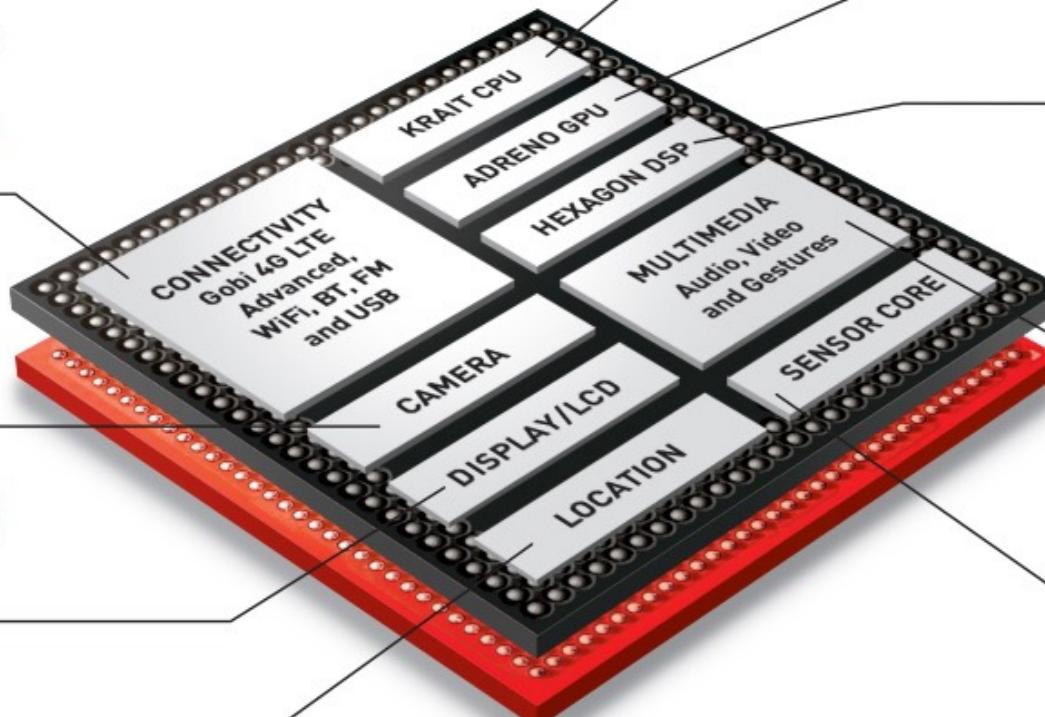
SNAPDRAGON 805 PROCESSOR

Stay connected and stream large files fast with industry leading connectivity, including the world's most advanced 4G LTE and VIVE™ 2-stream 802.11ac Wi-Fi

Capture sharper photos, even in low light, with the mobile industry's first dual ISP

Enjoy Ultra HD resolution content on Ultra HD-capable mobile devices and Ultra HD TVs with the Snapdragon Display Processor

Find your way outdoors and indoors with iZat GNSS with support for GPS, Glonass and BeiDou constellations



Faster performance and more multitasking with Krait 450 CPU at up to 2.7 GHz

Console quality gaming with new generation Adreno 420 GPU

More power-efficient apps and system processing with the Hexagon™ QDSP6

Capture and play back Ultra HD video and enjoy 7.1 surround sound on the go or at home with advanced video and audio engines

Get more use and greater accuracy from sensor-intensive apps with the dedicated Snapdragon Sensor Engine

Cerebras: Wafer-Scale Engine 2

Cerebras WSE-2 The Largest Chip Ever Built

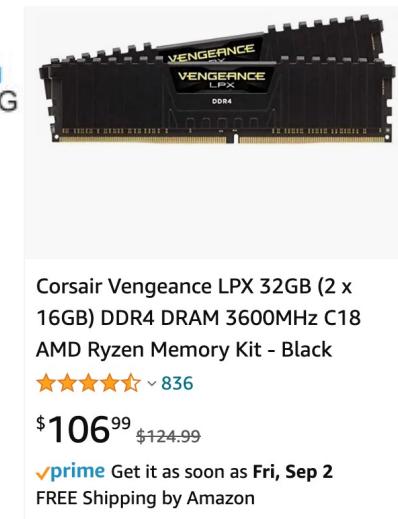
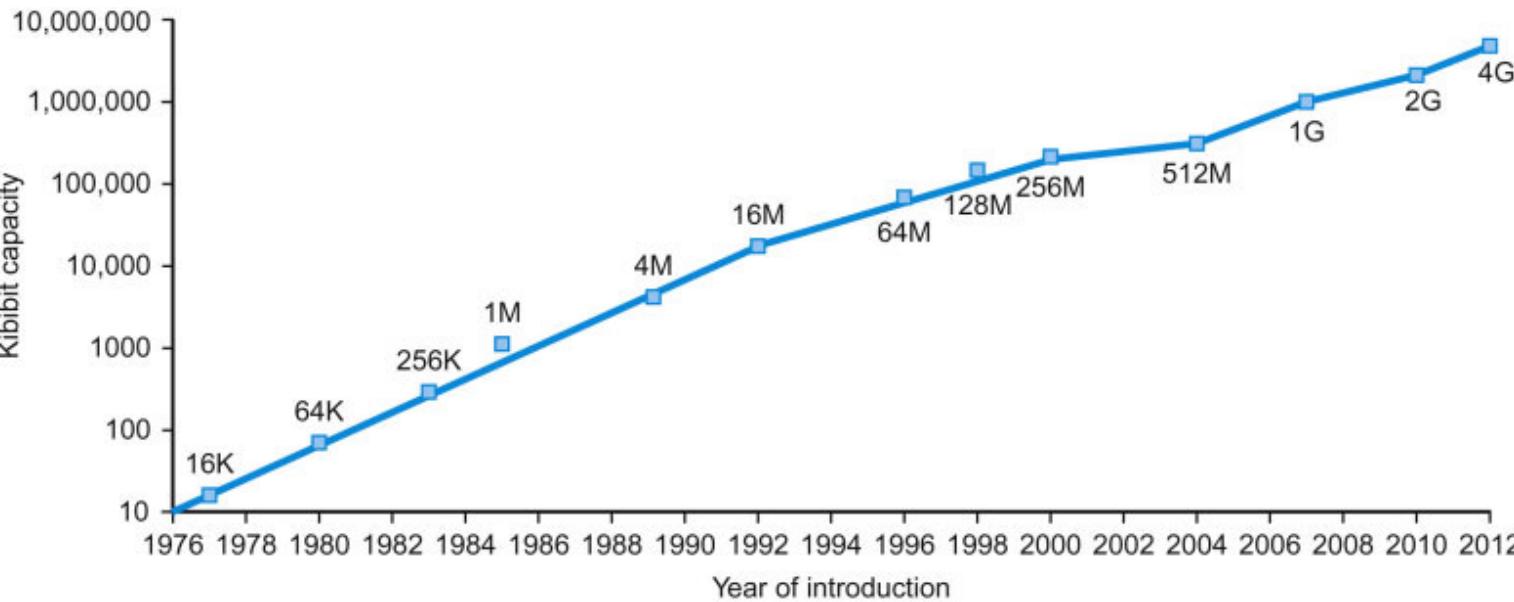
46,225	mm ² silicon
2.6	Trillion transistors
850,000	AI optimized cores
40	Gigabytes on chip memory
20	Petabyte/s memory bandwidth
220	Petabit/s fabric bandwidth
1.2	Terabit/s ingest bandwidth
7nm	Process technology at TSMC



Apple M2 → 5 nm

ASML → 2 nm

Technology Trends: Memory Capacity



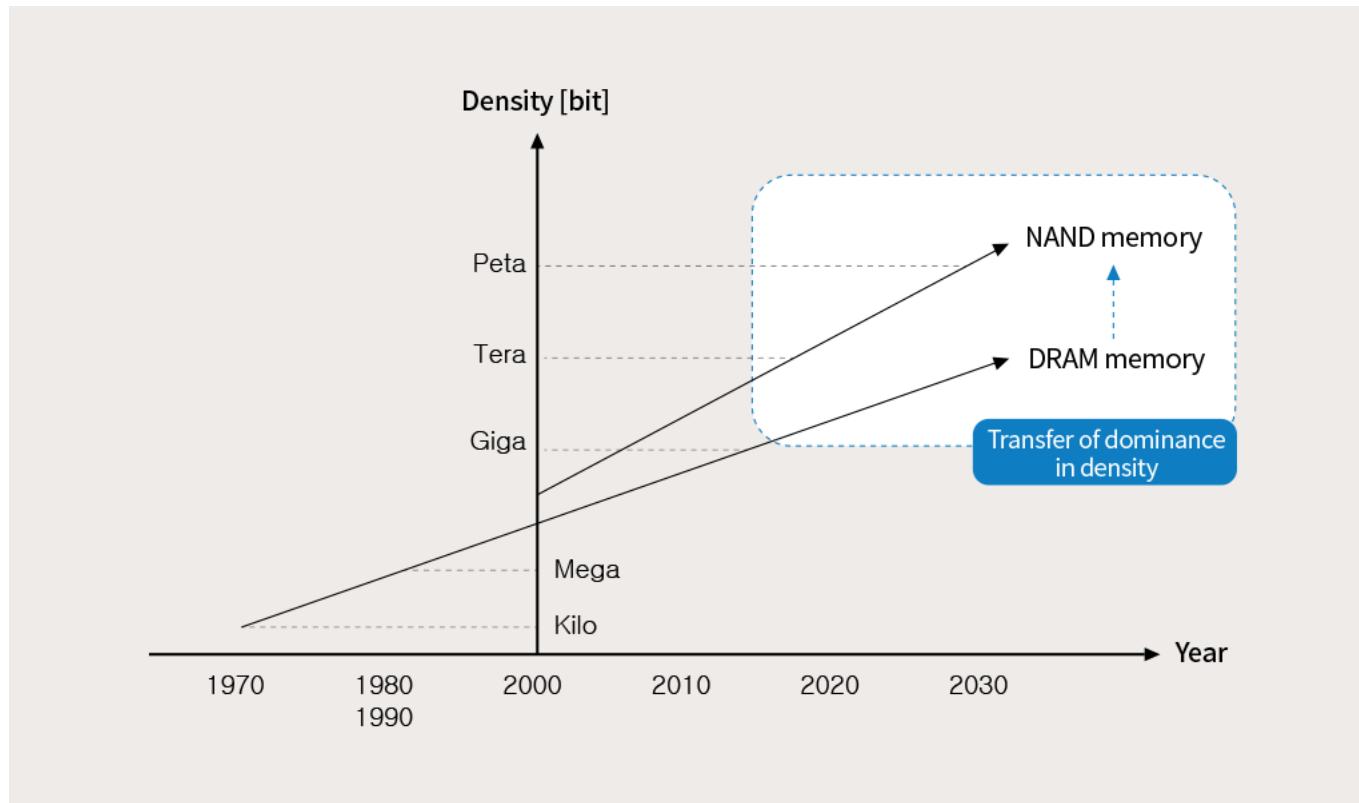
2022

FIGURE 1.11 Growth of capacity per DRAM chip over time. 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.

Technology Trends: Memory Capacity (Future)



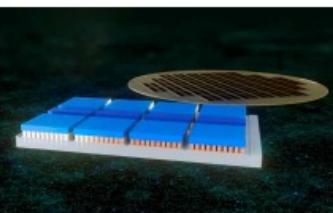
Is hardware research dead?

Relentlessly Pursuing Moore's Law

Advancing and Accelerating Computing

Components Research, the research group of Intel Technology Development, is responsible for delivering revolutionary process and packaging technologies that extend Moore's Law and enable Intel products and services.

At the 67th Annual IEEE International Electron Devices Meeting, Components Research is presenting key breakthroughs in three areas of research for delivering the fundamental building blocks for more powerful computing well into the future:



Delivering More Transistors

Essential scaling technologies for delivering more transistors include making them faster and smaller, so we can deliver millions more per square area and deliver them as tiles, or chiplets, via advanced packaging.

- Interconnect density improvement in packaging
- Area improvement with 3D transistor stacking
- Super thin materials for future scaling



Bringing New Capabilities to Silicon

As we enable more powerful computing through scaling, we need to stretch the limits of silicon and integrate new materials so we can deliver power more efficiently and meet greater demands for memory.

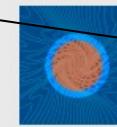
- World's first integration of GaN-based power switches
- Record FeRAM speed and endurance



Embracing the Quantum Realm

We are exploring entirely new concepts in physics that may one day revolutionize computing by potentially replacing classic transistors, enabling even greater performance and power efficiencies.

- Magnetoelectric spin-orbit logic
- Spin-torque devices
- 300mm qubit process flows



Spintronics

Ferroelectric RAM

Learn more at <https://intel.ly/31PxF9p>

© Intel Corporation. Intel, the Intel logo and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.

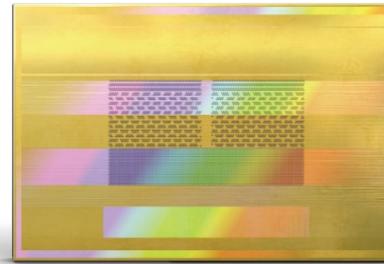
AI Computing Comes to Memory Chips

Samsung will double performance of neural nets with processing-in-memory

BY SAMUEL K. MOORE

John von Neumann's original computer architecture, where logic and memory are separate domains, has had a good run. But some companies are betting that it's time for a change.

In recent years, the shift toward more parallel processing and a massive increase in the size of neural networks mean processors need to access more data from memory more quickly. And yet "the performance gap between DRAM and processor is wider than ever," says [Joungho Kim](#), an expert in 3D memory chips at Korea Advanced Institute of Science and Technology, in Daejeon, and an IEEE Fellow. The von Neumann architecture has become the von Neumann bottleneck.

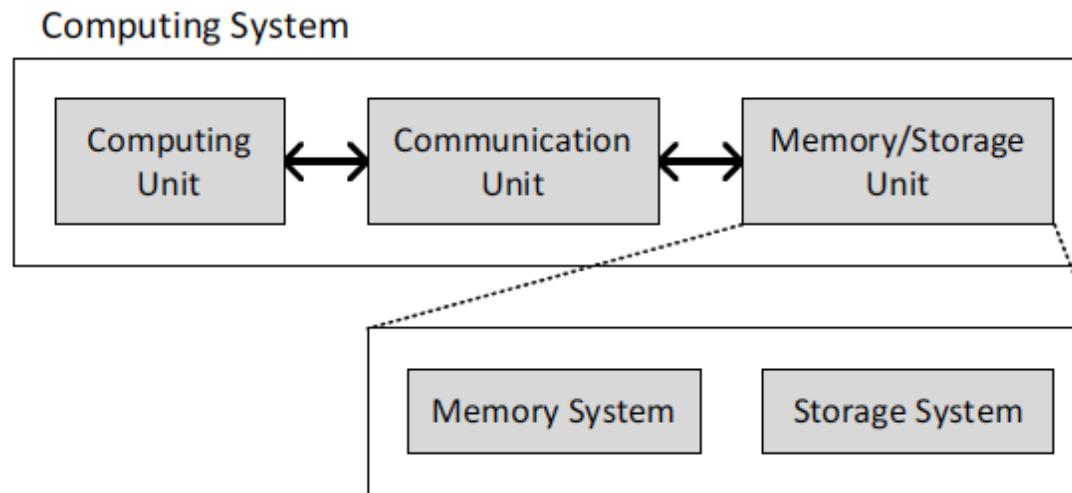


Samsung

Samsung added AI compute cores to DRAM memory dies to speed up machine learning.

Don't Panic

- In this course, we will only cover the basics
- So, when we say a computer, we mean a device that can compute, communicate, and store some information



The things we will learn in this course can be used almost in device with a processor and a memory!

Course Objectives

- To learn the organizational paradigms that determine the capabilities and performance of computer systems
- To understand the interactions between the computer's architecture and its software so that
 - **future software designers** (compiler writers, operating system designers, database programmers, ...) can achieve the best cost-performance trade-offs
 - **future architects** understand the effects of their design choices on software applications

Read Chapter 1, Sections 1-12, before our next class (9/7)