Digital Design & Computer Arch.

Lecture 1: Introduction: Fundamentals, Transistors, Gates

Prof. Onur Mutlu

ETH Zürich

Spring 2025

20 February 2025

Brief Self Introduction

- Onur Mutlu
  - Full Professor @ ETH Zurich ITET (INFK), since Sept 2015
  - Strecker Professor @ Carnegie Mellon University ECE (CS), 2009-2016, 2016-…
  - Started the Comp Arch Research Group @ Microsoft Research, 2006-2009
  - Worked @ Google, VMware, Microsoft Research, Intel, AMD, Stanford
  - PhD in Computer Engineering from University of Texas at Austin in 2006
  - BS in Computer Engineering & Psychology from University of Michigan in 2000
  - https://people.inf.ethz.ch/omutlu/ omutlu@gmail.com

- Research and Teaching in:
  - **Computer architecture, systems, hardware security, bioinformatics**
  - Memory and storage systems
  - Robust & dependable hardware systems: security, safety, predictability, reliability
  - Hardware/software cooperation
  - New computing paradigms; architectures with emerging technologies/devices
  - Architectures for bioinformatics, genomics, health, medicine, AI/ML
  - …

My Co-Instructor

- Mohammad Sadrosadati
  - Senior Researcher and Lecturer @ SAFARI Research Group, ETHZ
  - PhD in Computer Engineering from Sharif University of Technology in 2019
  - MS in Computer Engineering from Sharif University of Technology in 2014
  - BS in Computer Engineering from Sharif University of Technology in 2012
  - mohammad.sadrosadati@safari.ethz.ch

- Research & Teaching Areas
  - Computer Architecture
  - Memory/Storage Systems
  - Near-Data Processing
  - Heterogeneous System Architecture
  - Bioinformatics
  - Interconnection Network

Head Teaching & Lab Assistants

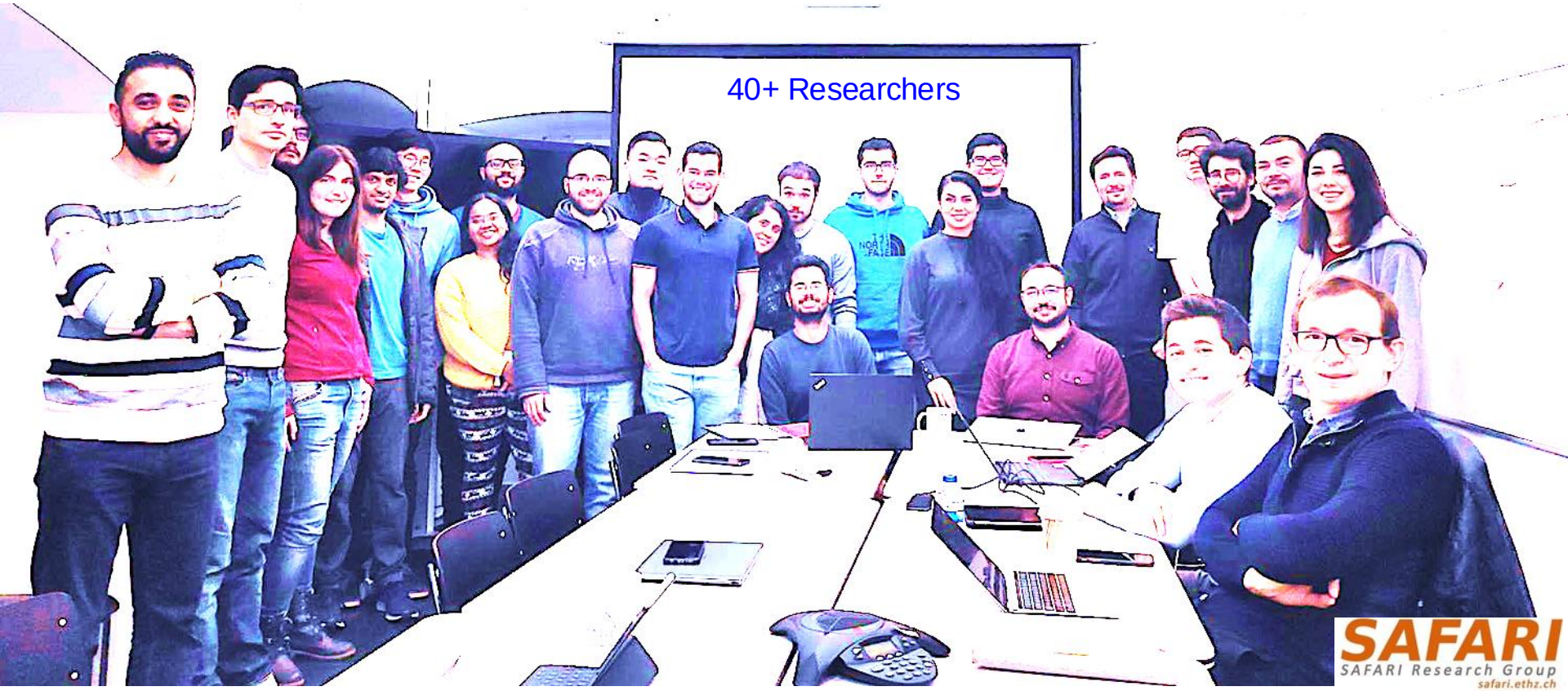
- Dr. Konstantina Koliogeorgi
  - Head Teaching Assistant
  - Senior Researcher and Lecturer @ SAFARI
  - PhD, National Technical University of Athens, 2023

- Ataberk Olgun
  - Head Lab Assistant
  - PhD Student @ SAFARI

SAFARI Research Group

Computer architecture, HW/SW, systems, bioinformatics, security, memory

https://safari.ethz.ch/safari-newsletter-april-2020/



Think BIG, Aim HIGH!

SAFARI

https://safari.ethz.ch

SAFARI Newsletter January 2021 Edition

- https://safari.ethz.ch/safari-newsletter-january-2021/

SAFARI
SAFARI Research Group

Newsletter
January 2021

Think Big, Aim High, and
Have a Wonderful 2021!

Dear SAFARI friends,

Happy New Year! We are excited to share our group highlights with you in this second edition of the SAFARI newsletter (You can find the first edition from April 2020 here). 2020 has

SAFARI Newsletter July 2024 Edition

- **https://safari.ethz.ch/safari-newsletter-july-2024/**

SAFARI Introduction & Research

Computer architecture, HW/SW, systems, bioinformatics, security, memory

SAFARI
SAFARI Research Group
safari.ethz.ch

Think BIG, Aim HIGH!

SAFARI

https://www.youtube.com/watch?v=mV2OuB2djEs

SAFARI PhD and Post-Doc Alumni

- https://safari.ethz.ch/safari-alumni/

- Can Firtina (ETH Zurich)
- Lukas Breitwieser (CERN), Best artifact award at PPoPP 2023
- A. Giray Yaglikci (ETH Zurich), PACT 2023 SRC Winner, Intel Hardware Security Academic Award Finalist 2021, HOST PhD Competition Finalist 2024
- Hasan Hassan (Rivos), EDAA Outstanding Dissertation Award 2023; S&P 2020 Best Paper Award, 2020 Pwnie Award, IEEE Micro TP HM 2020
- Christina Giannoula (Univ. of Toronto), NTUA Best Dissertation Award 2023
- Minesh Patel (Rutgers, Asst. Prof.), DSN Carter Award Best Thesis 2022; ETH Medal 2023; MICRO'20 & DSN'20 Best Paper Awards; ISCA HoF 2021
- Damla Senol Cali (Bionano Genomics), SRC TECHCON 2019 Best Student Presentation Award; RECOMB-Seq 2018 Best Poster Award
- Nastaran Hajinazar (Intel)
- Gagandeep Singh (AMD/Xilinx), FPL 2020 Best Paper Award Finalist
- Amirali Boroumand (Stanford Univ → Google), SRC TECHCON 2018 Best Presentation Award
- Jeremie Kim (Apple), EDAA Outstanding Dissertation Award 2020; IEEE Micro Top Picks 2019; ISCA/MICRO HoF 2021
- Nandita Vijaykumar (Univ. of Toronto, Assistant Professor), ISCA Hall of Fame 2021
- Kevin Hsieh (Microsoft Research, Senior Researcher)
- Justin Meza (Facebook), HiPEAC 2015 Best Student Presentation Award; ICCD 2012 Best Paper Award
- Mohammed Alser (ETH Zurich), IEEE Turkey Best PhD Thesis Award 2018
- Yixin Luo (Google), HPCA 2015 Best Paper Session
- Kevin Chang (Facebook), SRC TECHCON 2016 Best Student Presentation Award
- Rachata Ausavarungnirun (KMUNTB, Assistant Professor), NOCS 2015 and NOCS 2012 Best Paper Award Finalist
- Gennady Pekhimenko (Univ. of Toronto, Assistant Professor), ISCA Hall of Fame 2021; ASPLOS 2015 SRC Winner
- Vivek Seshadri (Microsoft Research, Principal Researcher)
- Donghyuk Lee (NVIDIA Research, Senior Researcher), HPCA Hall of Fame 2018
- Yoongu Kim (Software Robotics → Google), IFIP JCL Award'24, TCAD'19 Top Pick Award; IEEE Micro Top Picks'10; HPCA'10 Best Paper Session
- Lavanya Subramanian (Intel Labs → Facebook)

- Samira Khan (Univ. of Virginia, Assistant Professor), HPCA 2014 Best Paper Session
- Saugata Ghose (Univ. of Illinois, Assistant Professor), DFRWS-EU 2017 Best Paper Award
- Jawad Haj-Yahya (Huawei Research Zurich, Principal Researcher)
- Lois Orosa (Galicia Supercomputing Center, Director)
- Jisung Park (POSTECH, Assistant Professor)
- Gagandeep Singh (AMD/Xilinx, Researcher)
- Juan Gomez-Luna (NVIDIA, Researcher), ISPASS 2023 Best Paper Session
- Mohammed Alser (Georgia State Univ. Assistant Professor), IEEE Turkey Best PhD Thesis Award 2018

An Interview on Computing Futures

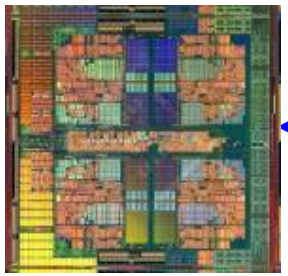
https://www.youtube.com/watch?v=8ffSEKZhmvo

Suggestions on Education, Research, Growth

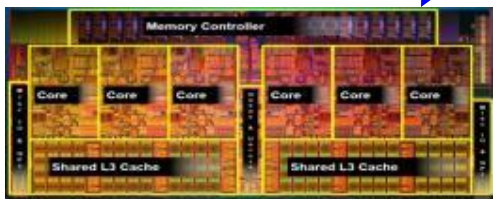
https://www.youtube.com/watch?v=83tlorht7Mc&list=PL5Q2soXY2Zi8D_5MGV6EnXEJHnV2YFBJl&index=54

SAFARI Research Group: Current Mission

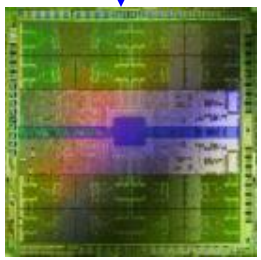
Computer architecture, HW/SW, systems, bioinformatics, security



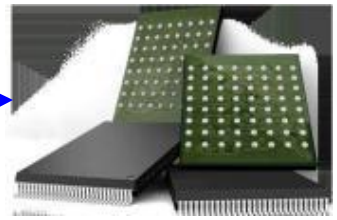
Heterogeneous Processors and Accelerators



Processing in Memories



ML/AI, Vision, Genomics, Biological Signal Processing



Storage, Sensors, Memory, Interconnect with Processing Capabilities + New Paradigms for Computation

https://safari.ethz.ch

SAFARI
SAFARI Research Group
safari.ethz.ch

Enable fundamentally better computers

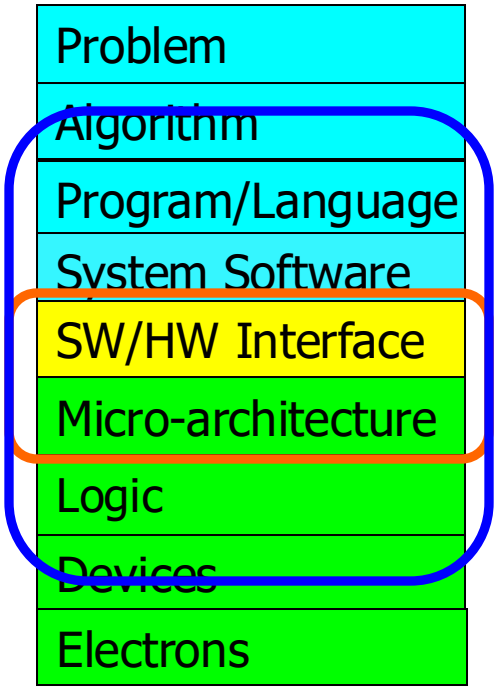
https://safari.ethz.ch/safari-newsletter-july-2024/

Major Current Research Topics

- Fundamentally Robust (Secure/Reliable/Safe) Architectures
- Fundamentally Energy-Efficient Architectures
  - Memory-centric (Data-centric) Architectures
- Fundamentally Low-Latency and Predictable Architectures
- Fundamentally Intelligent and Evolving Architectures
  - ML/AI-Assisted (Data-driven) and Data-aware Architectures
- Architectures for ML/AI, Genomics, Medicine, Health, …

The Transformation Hierarchy

Computer Architecture (expanded view)



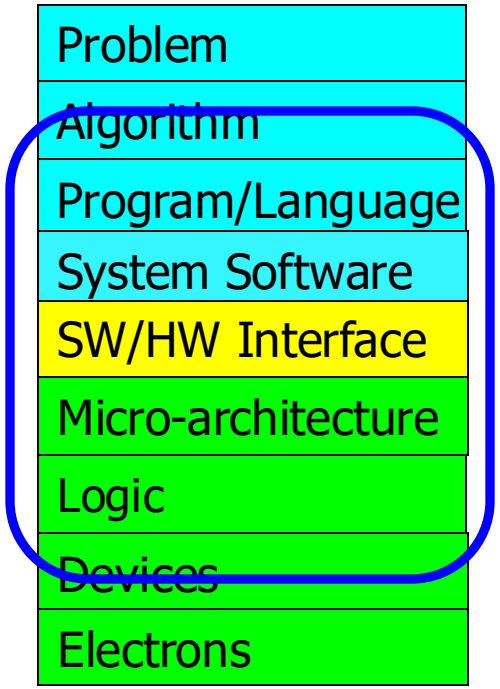
Computer Architecture (narrow view)

Approach: Cross Layer Design

To achieve the highest efficiency, performance, robustness:

we must take the expanded view

of computer architecture



Co-design across the hierarchy: Algorithms to devices

Specialize as much as possible within the design goals

Broad research spanning applications, systems, software, logic, circuits with architecture at the center

Principle: Teaching and Research

...

Teaching drives Research

Research drives Teaching

...

Accessing All Our Courses

SAFARI Think Big, Aim High | Home | People | Courses | News | Research | Publications | Tools | Work with us | Contact us

to state-of-the-art computer architecture classes benefit from our online classes (see our YouTube channels here).

Spring 2025:
Fundamentals of Computer Architecture
Digital Design and Computer Architecture
Seminar in Computer Architecture
SAFARI **Project & Seminars** courses

Fall 2024:
Computer Architecture
Seminar in Computer Architecture
SAFARI **Project & Seminars** courses

Spring 2024:
Digital Design and Computer Architecture
Seminar in Computer Architecture
SAFARI **Project & Seminars** courses

Fall 2023:
Computer Architecture
Seminar in Computer Architecture
SAFARI **Project & Seminars** courses

Spring 2023:
Digital Design and Computer Architecture
Seminar in Computer Architecture
SAFARI **Project & Seminars** courses

Onur Mutlu Lectures
47.1K subscribers

Subscribe to our newsletter

First name *

Last name *

Email *

Subscribe!

https://safari.ethz.ch/courses

ETH zürich

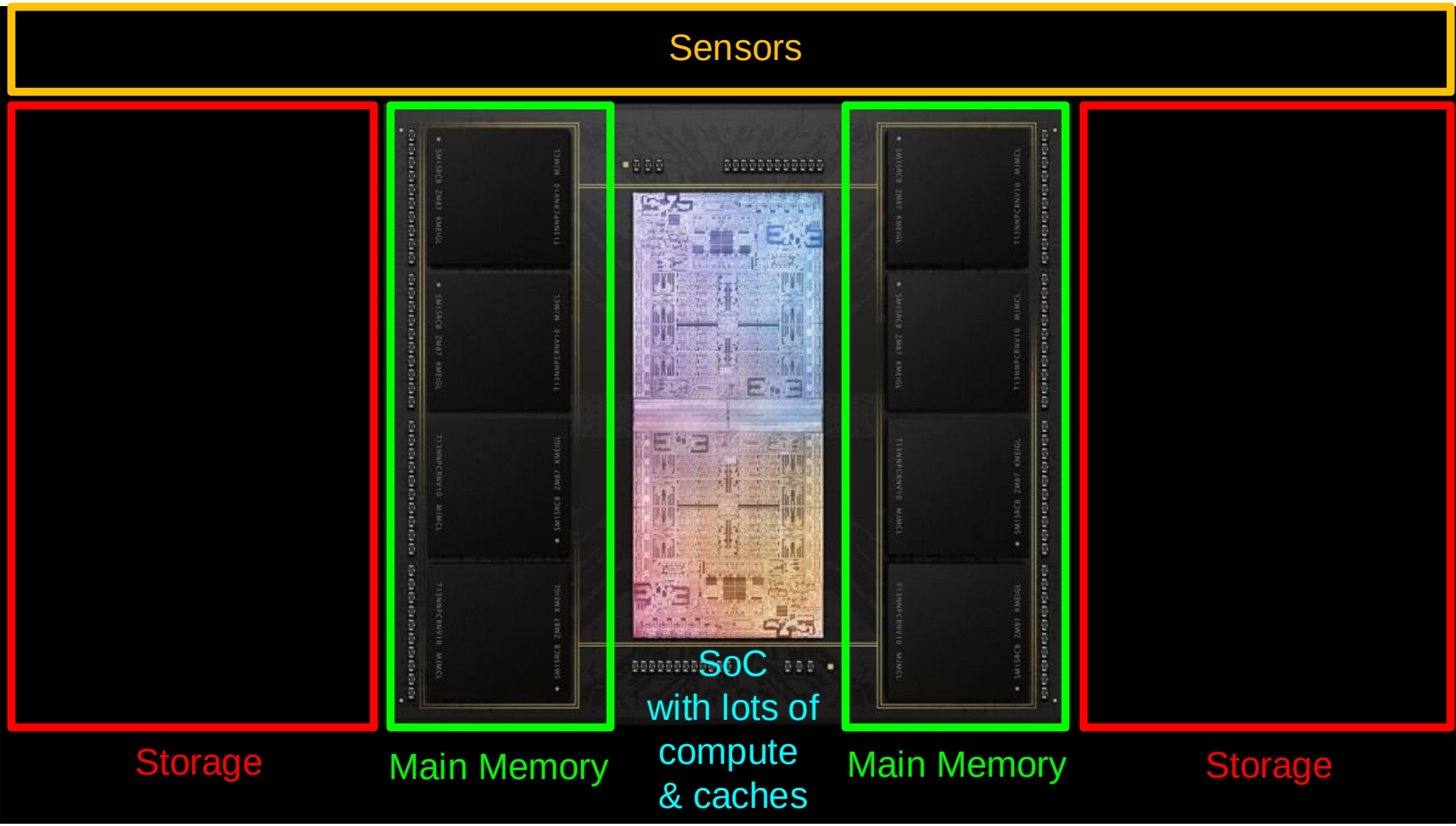
Basic Goals & Structure of DDCA

What Will We Learn in This Course?

How Computers Work

(from the ground up)

We Will Study How Something Like This Works



Apple M1 Ultra System (2022)

https://www.gsmarena.com/apple_announces_m1_ultra_with_20core_cpu_and_64core_gpu-news-53481.php

Major High-Level Goals of This Course

- In Digital Design & Computer Architecture

- Understand the basics
- Understand the principles (of design)
- Understand the precedents

- Based on such understanding:
  - learn how a modern computer works underneath
  - evaluate tradeoffs of different designs and ideas
  - implement a principled design (a simple microprocessor)
  - learn to systematically debug increasingly complex systems
  - Hopefully enable you to develop novel, out-of-the-box designs

- The focus is on basics, principles, precedents, and how to use them to create/implement good designs

Why These Goals?

- Because you are here for a Computer Science degree!

- Regardless of your future direction, learning the principles of digital design & computer architecture will be useful to
  - design better hardware
  - design better software
  - design better systems
  - make better tradeoffs in design
  - understand why computers behave the way they do
  - solve problems better
  - think "in parallel"
  - think critically
  - …

DDCA Course Components

- **Lectures** (understanding concepts)
- **Readings** (reinforcing & going deeper)
- **Homeworks** (problem solving preparation)
- **Labs** (hands-on experience in some concepts)
- **Exam** (test of understanding)
- **Extra Credit Assignments** (fundamental and simple)

- In all, you have freedom to adapt to your learning style

- My advice: Focus on learning & scholarship & understanding

https://safari.ethz.ch/ddca/spring2025/

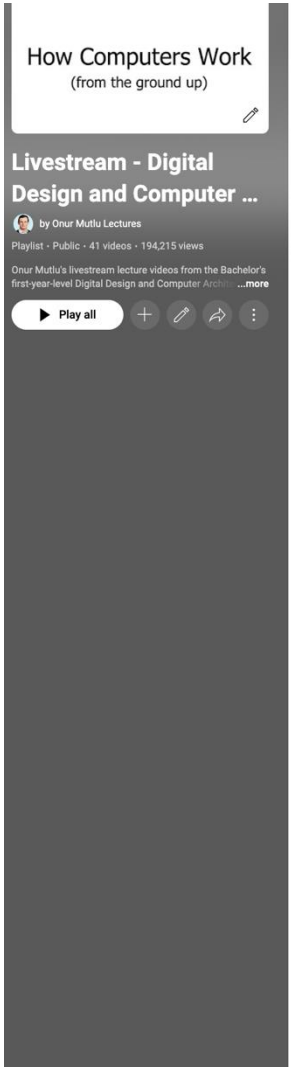
Learning & Exam

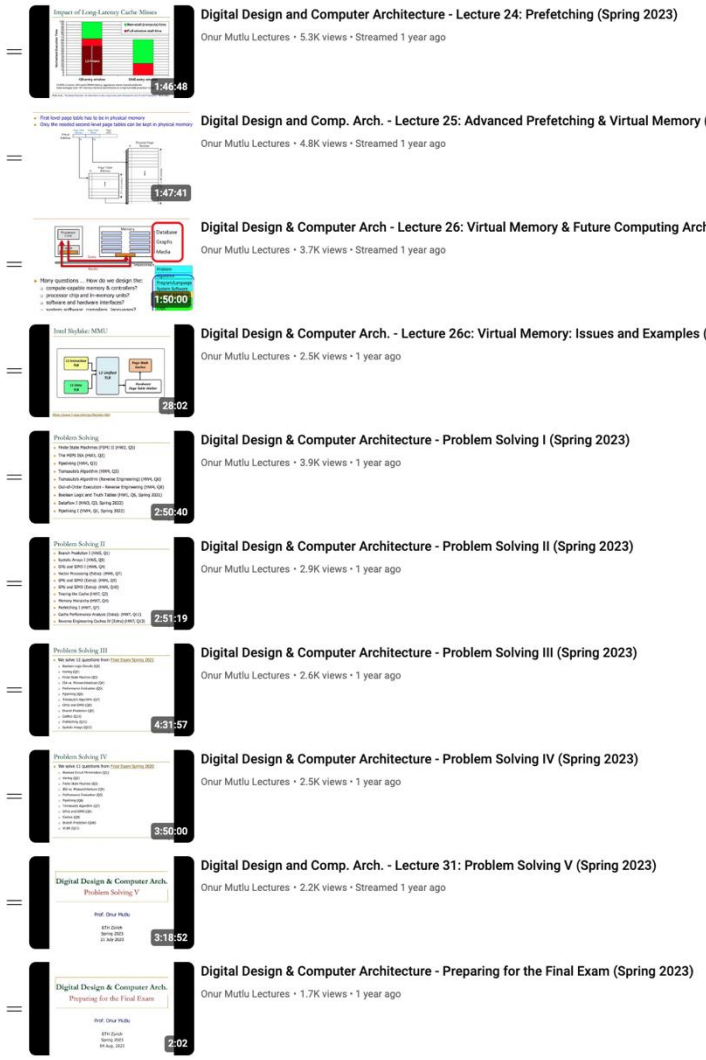
- We will enable you to learn + prepare you for the exam

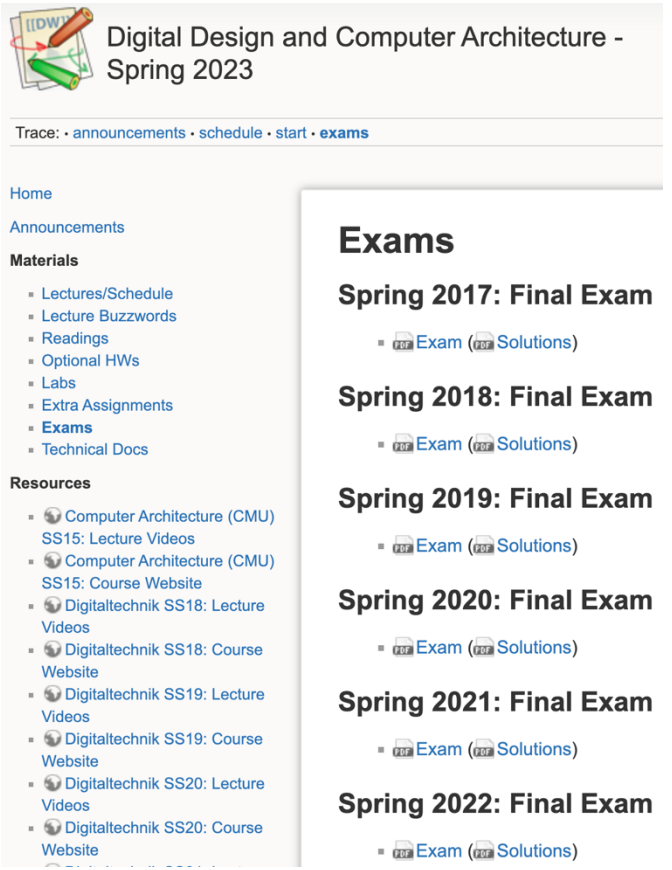
- My suggestions:
  - focus on understanding, learning, mastering the material
    - lectures, readings, labs, HWs all enable this and prepare you
  - reinforce problem solving skills with homeworks
  - do **not** worry about the exam while listening to lectures
    - most of you will pass this course (historically >80%)

- We will release a lot of material to help you with the exam
  - Problem solving sessions
  - Exam guidance
  - All past exams (and basic solutions) are already online

Problem Solving & Exam Review Sessions







https://safari.ethz.ch/digitaltechnik/spring2023

https://www.youtube.com/playlist?list=PL5Q2soXY2Zi-EImKxYYY1SZuGiOAOBKaf

Summary

- Learning is for life (never ends)
- Exam study is until you pass (ends, hopefully August 2025)

Focus on

learning and scholarship

How to Approach This Course

Learning experience

Long-term tradeoff analysis

Critical thinking & decision making

How to Approach This Course

Find and choose
the learning style
that works best for you

What Will We Learn in This Course?

What Will We Learn in This Course?

How Computers Work

(from the ground up)

What Will We Learn in This Course?

And Why We Care

Why Do We Have Computers?

Why Do We Do Computing?

Why Do We Do Computing?

To Solve Problems

Answer Reworded

To Gain Insight

Answer Extended

To Enable

a Better Life & Future

How Does a Computer Solve Problems?

How Does a Computer Solve Problems?

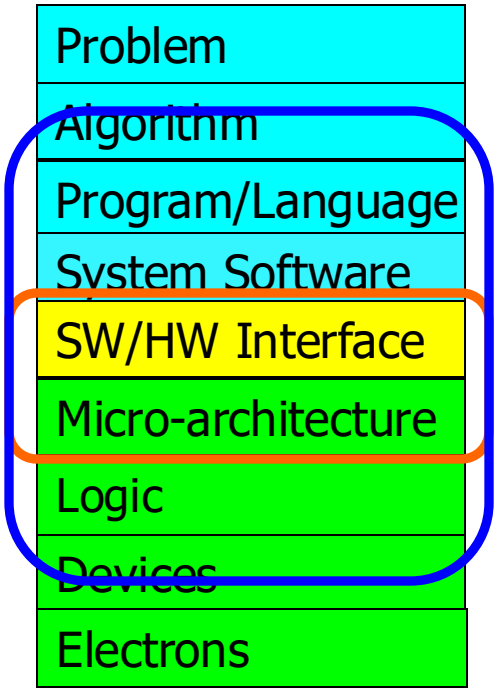
Orchestrating Electrons

In today's dominant technologies

How Do Problems Get Solved by Electrons?

The Transformation Hierarchy

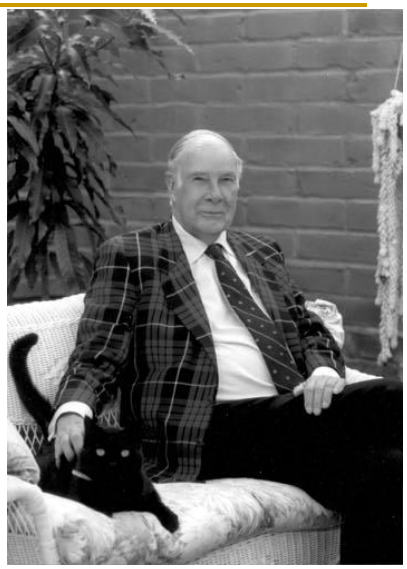
Computer Architecture (expanded view)



Computer Architecture (narrow view)

Levels of Transformation

"The purpose of computing is [to gain] insight" (*Richard Hamming*)
We gain and generate insight by solving problems
How do we ensure problems are solved by electrons?

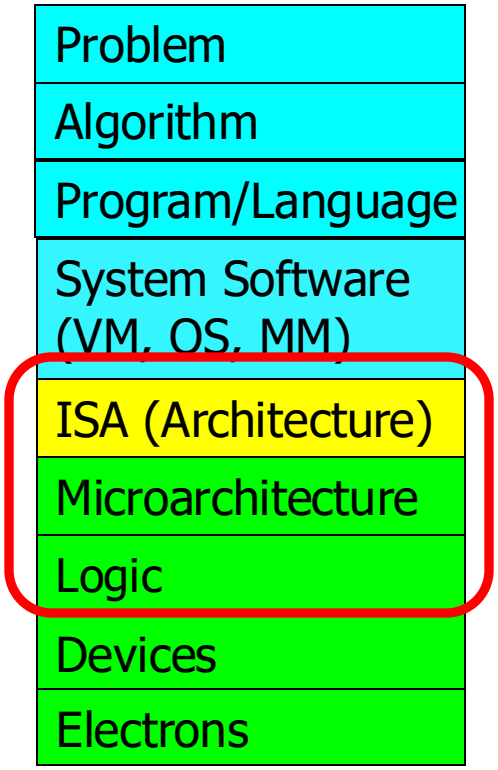


Algorithm

Step-by-step procedure that is **guaranteed to terminate** where **each step is precisely stated** and **can be carried out by a computer**

- **Finiteness**
- **Definiteness**
- **Effective computability**

Many algorithms for the same problem



ISA (Instruction Set Architecture)

Interface/contract between SW and HW.

What the programmer assumes hardware will satisfy.

Microarchitecture
An implementation of the ISA

Digital logic circuits
Building blocks of micro-arch (e.g., gates)

Computer Architecture

- is the science and art of designing computing platforms (hardware, interface, system SW, and programming model)

- to achieve a set of design goals
  - E.g., highest performance on earth on workloads X, Y, Z
  - E.g., longest battery life at a form factor that fits in your pocket with cost < $$$ CHF
  - E.g., best average performance across all known workloads at the best performance/cost ratio
  - …

  - Designing a supercomputer is different from designing a smartphone → But, many fundamental principles are similar

Different Platforms, Different Goals

Source: http://www.sia-online.org (semiconductor industry association)

Different Platforms, Different Goals

Source: https://iq.intel.com/5-awesome-uses-for-drone-technology/

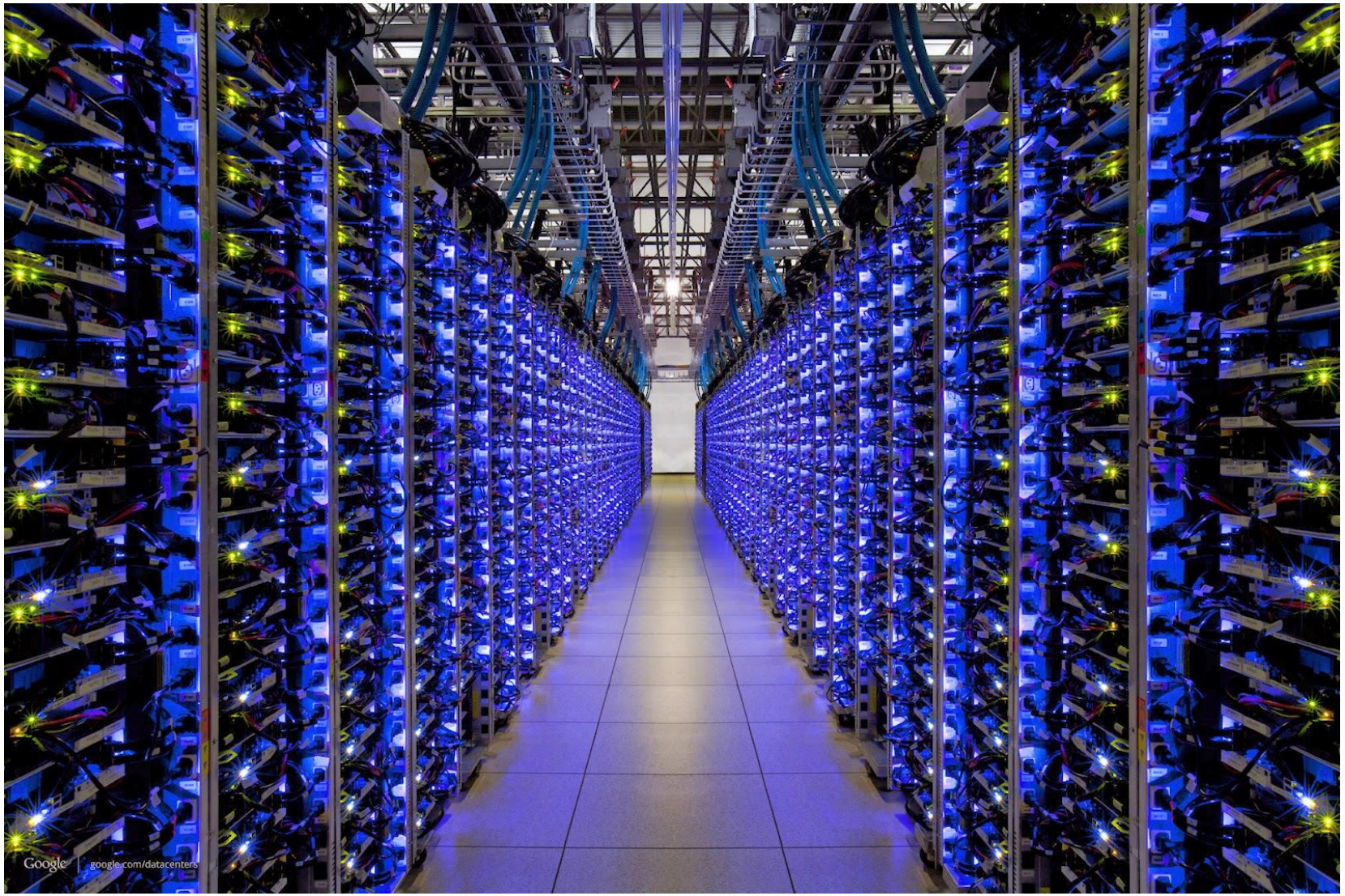
Different Platforms, Different Goals

Source: https://taxistartup.com/wp-content/uploads/2015/03/UK-Self-Driving-Cars.jpg

Different Platforms, Different Goals

Source: http://sm.pcmag.com/pcmag_uk/photo/g/google-self-driving-car-the-guts/google-self-driving-car-the-guts_dwx8.jpg

Different Platforms, Different Goals



Source: http://datacentervoice.com/wp-content/uploads/2015/10/data-center.jpg

Different Platforms, Different Goals

Source: https://fossbytes.com/wp-content/uploads/2015/06/Supercomputer-TIANHE2-china.jpg

Different Platforms, Different Goals

Source: https://www.itmagazine.ch/artikel/72401/Fugaku_Der_schnellste_Supercomputer_der_Welt.html

Different Platforms, Different Goals

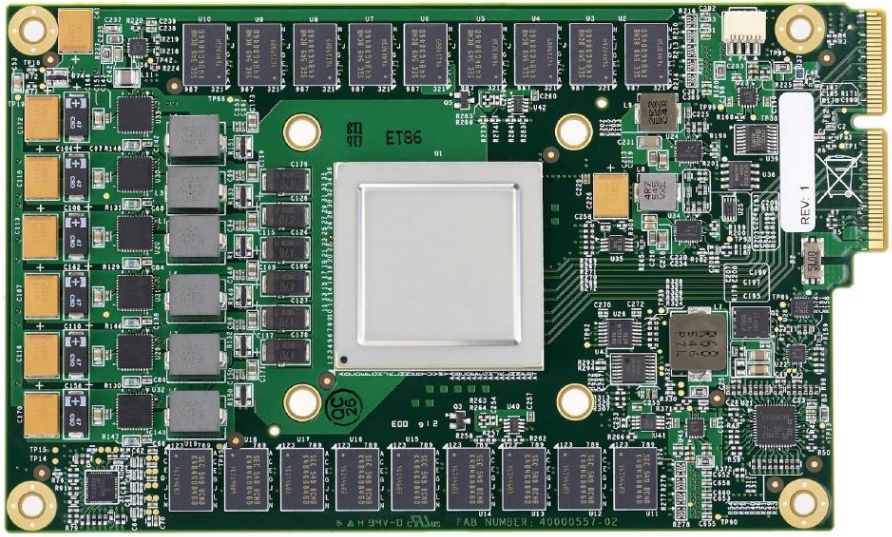


Figure 3. TPU Printed Circuit Board. It can be inserted in the slot for an SATA disk in a server, but the card uses PCIe Gen3 x16.

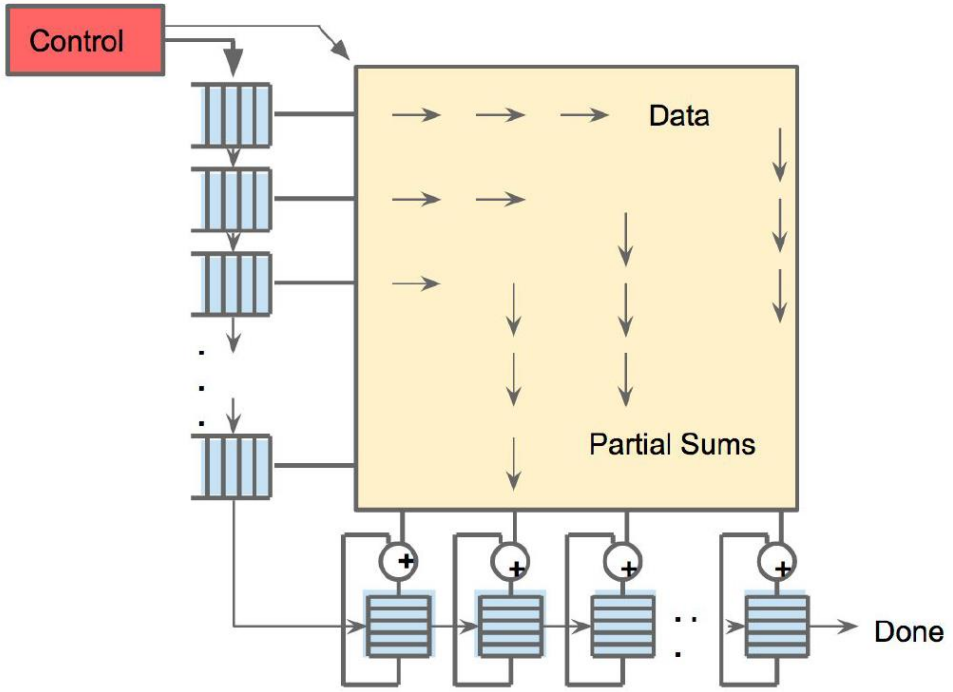
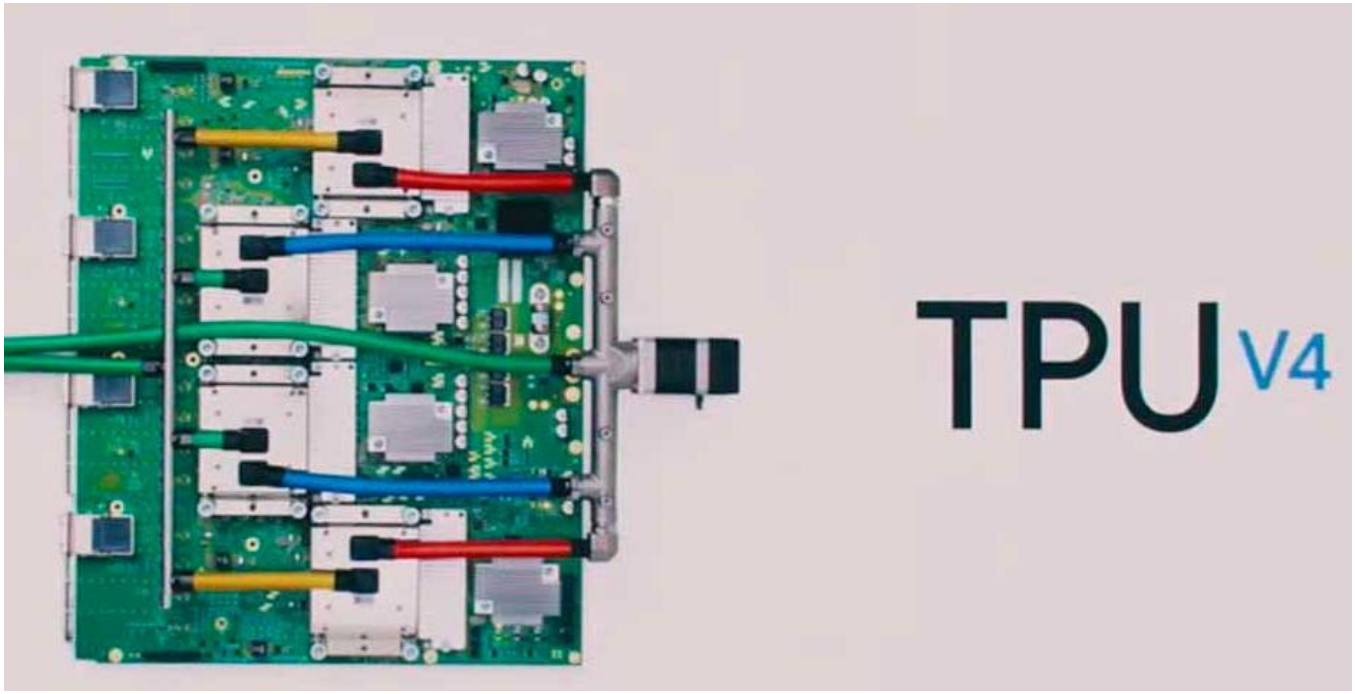


Figure 4. Systolic data flow of the Matrix Multiply Unit. Software has the illusion that each 256B input is read at once, and they instantly update one location of each of 256 accumulator RAMs.

Jouppi et al., "In-Datacenter Performance Analysis of a Tensor Processing Unit", ISCA 2017.

Different Platforms, Different Goals



New ML applications (vs. TPU3):

- Computer vision
- Natural Language Processing (NLP)
- Recommender system
- Reinforcement learning that plays Go

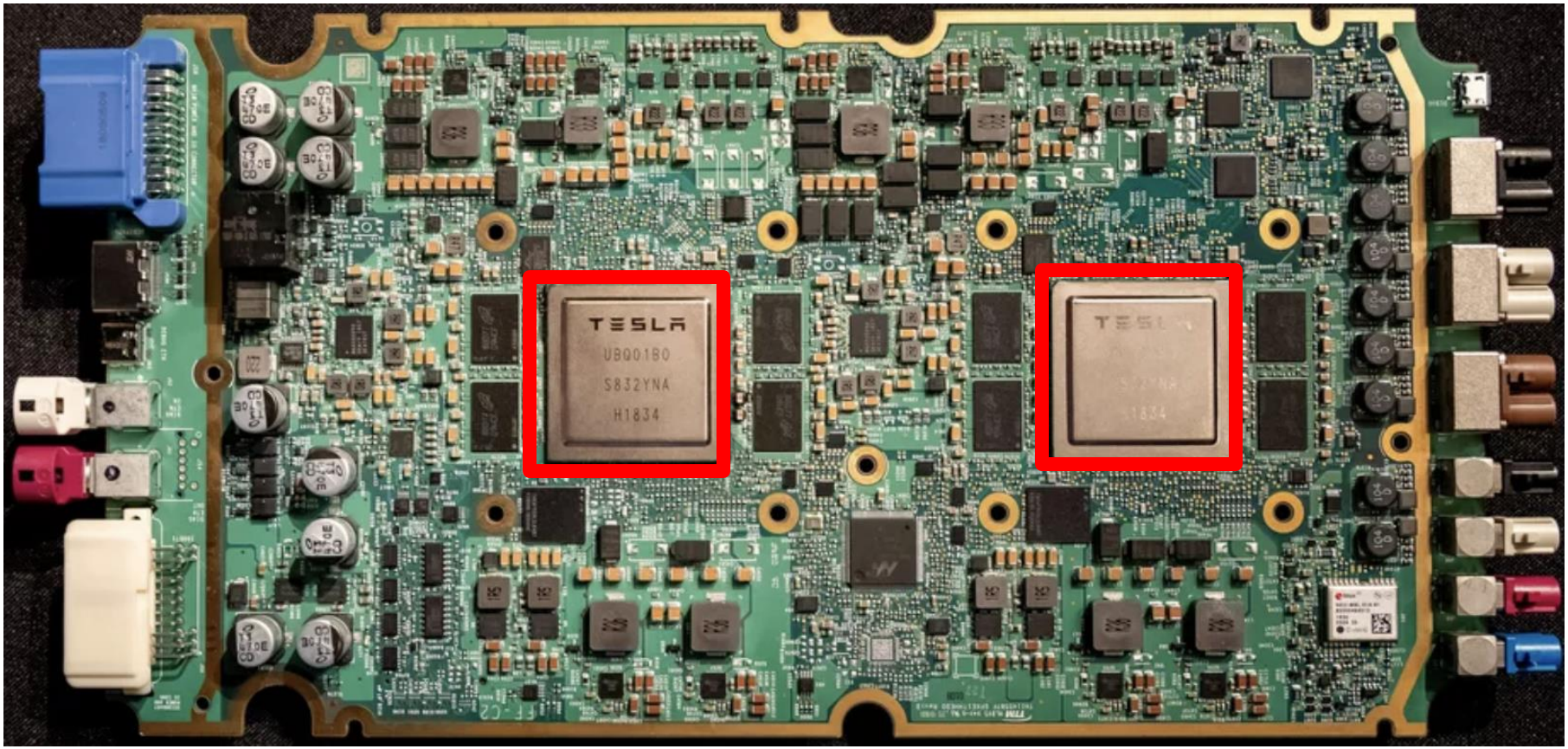
250 TFLOPS per chip in 2021
vs 90 TFLOPS in TPU3

1 ExaFLOPS per board

https://spectrum.ieee.org/tech-talk/computing/hardware/heres-how-googles-tpu-v4-ai-chip-stacked-up-in-training-tests

Different Platforms, Different Goals

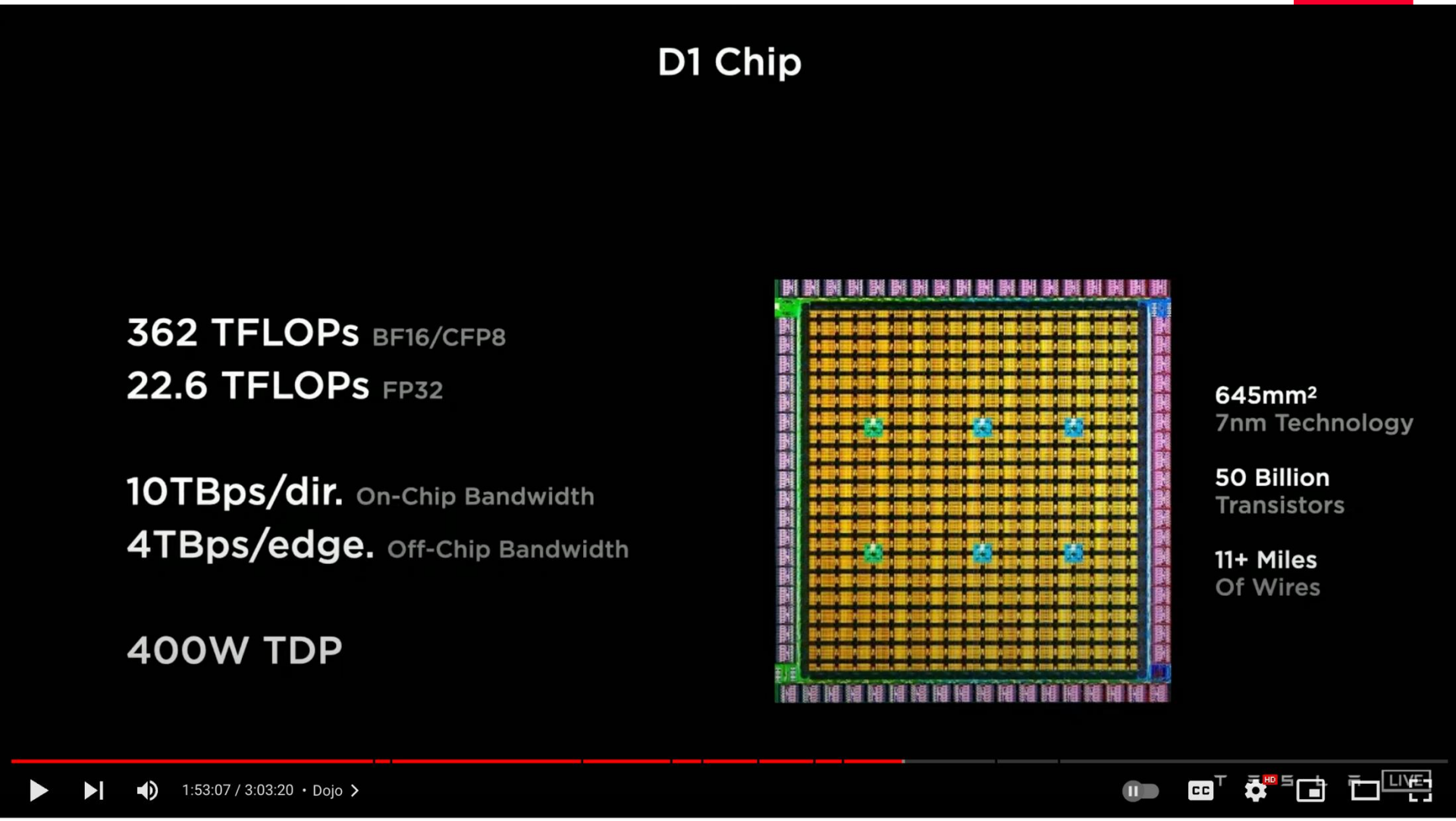
- ML accelerator: 260 mm^2, 6 billion transistors, 600 GFLOPS GPU, 12 ARM 2.2 GHz CPUs.
- Two redundant chips for better safety.



https://www.youtube.com/watch?v=j0z4FweCy4M

Different Platforms, Different Goals

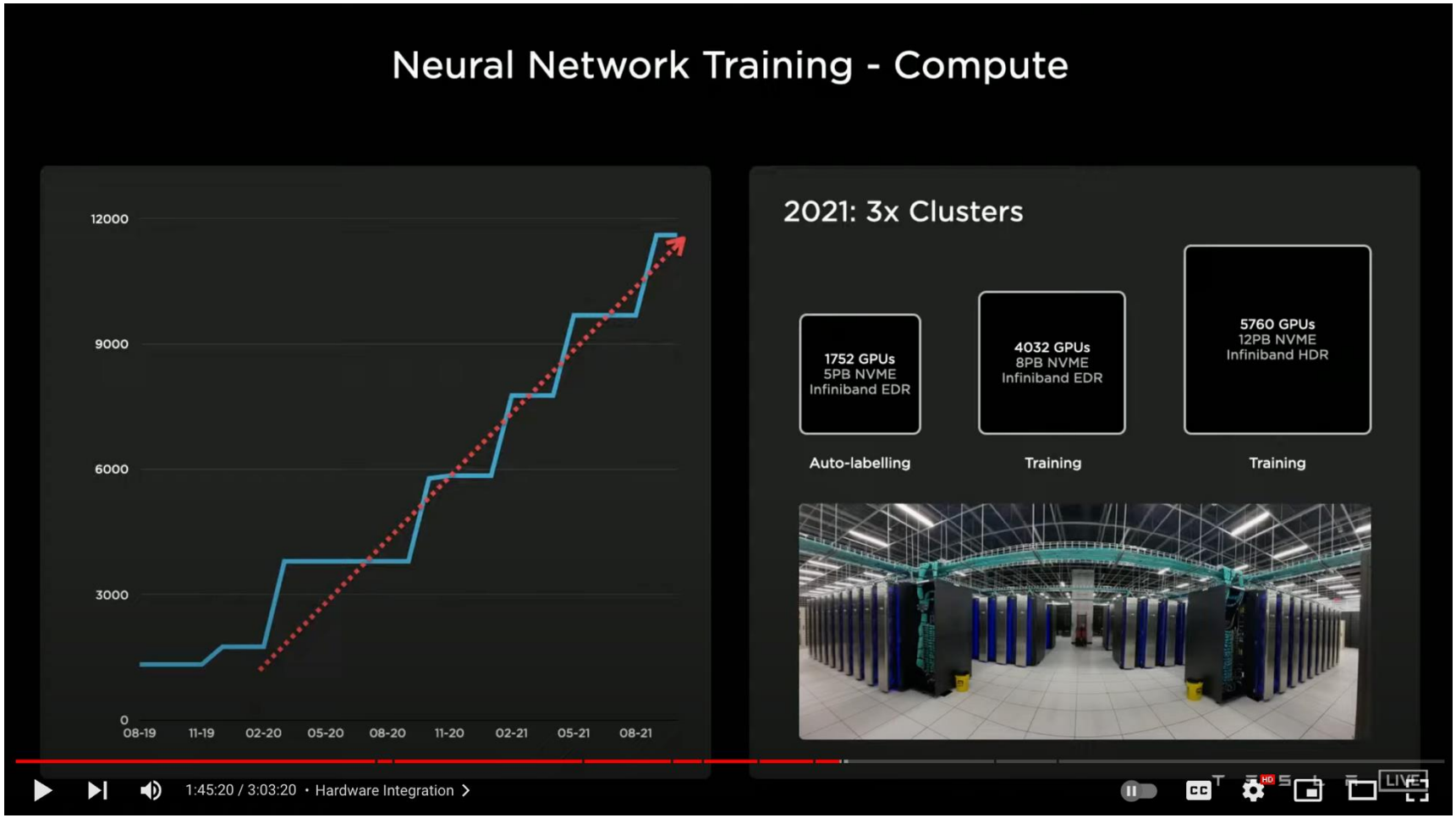
- Tesla Dojo Chip & System



https://www.youtube.com/watch?v=j0z4FweCy4M&t=6340s

Different Platforms, Different Goals

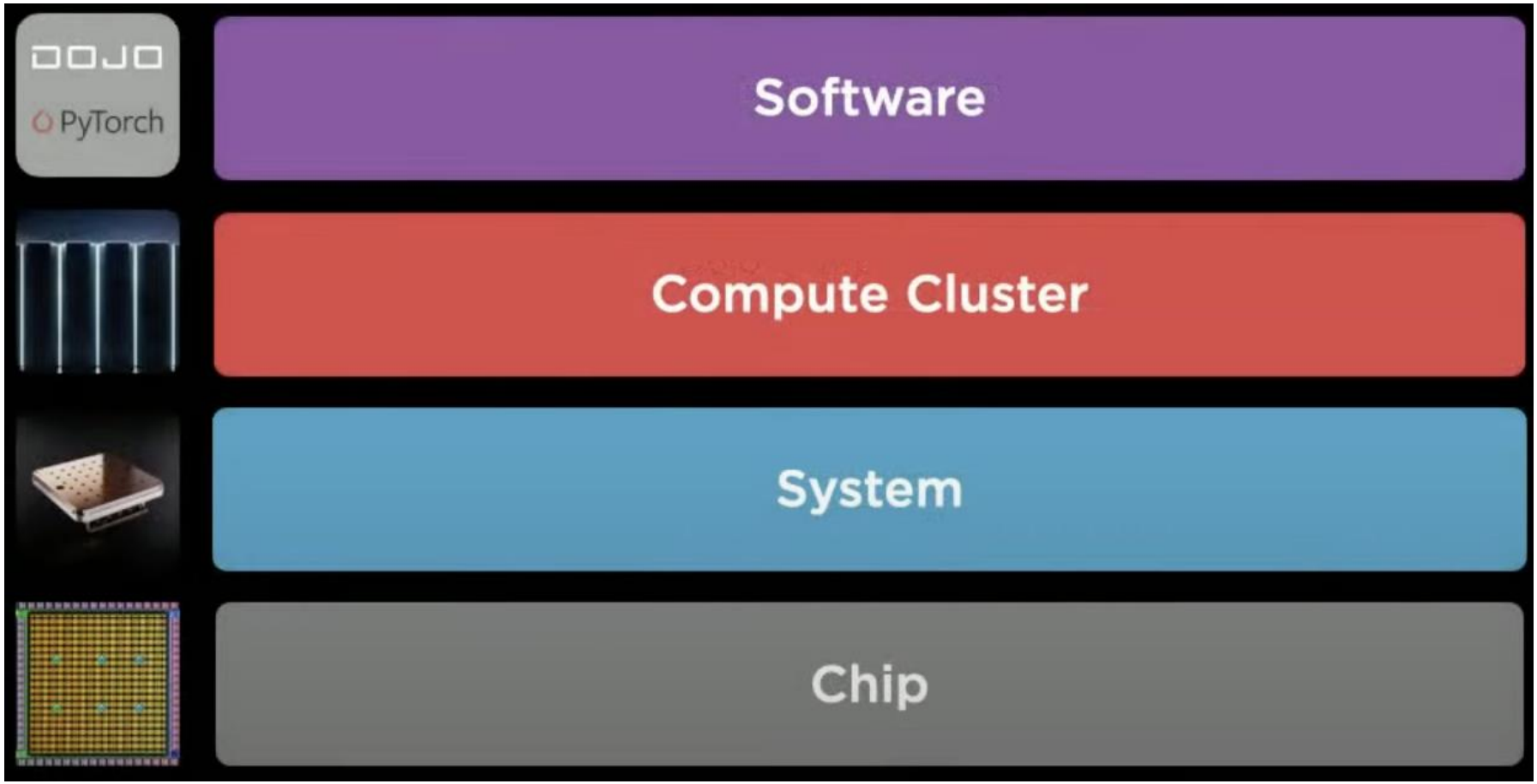
- Tesla Dojo Chip & System



https://www.youtube.com/watch?v=j0z4FweCy4M&t=6340s

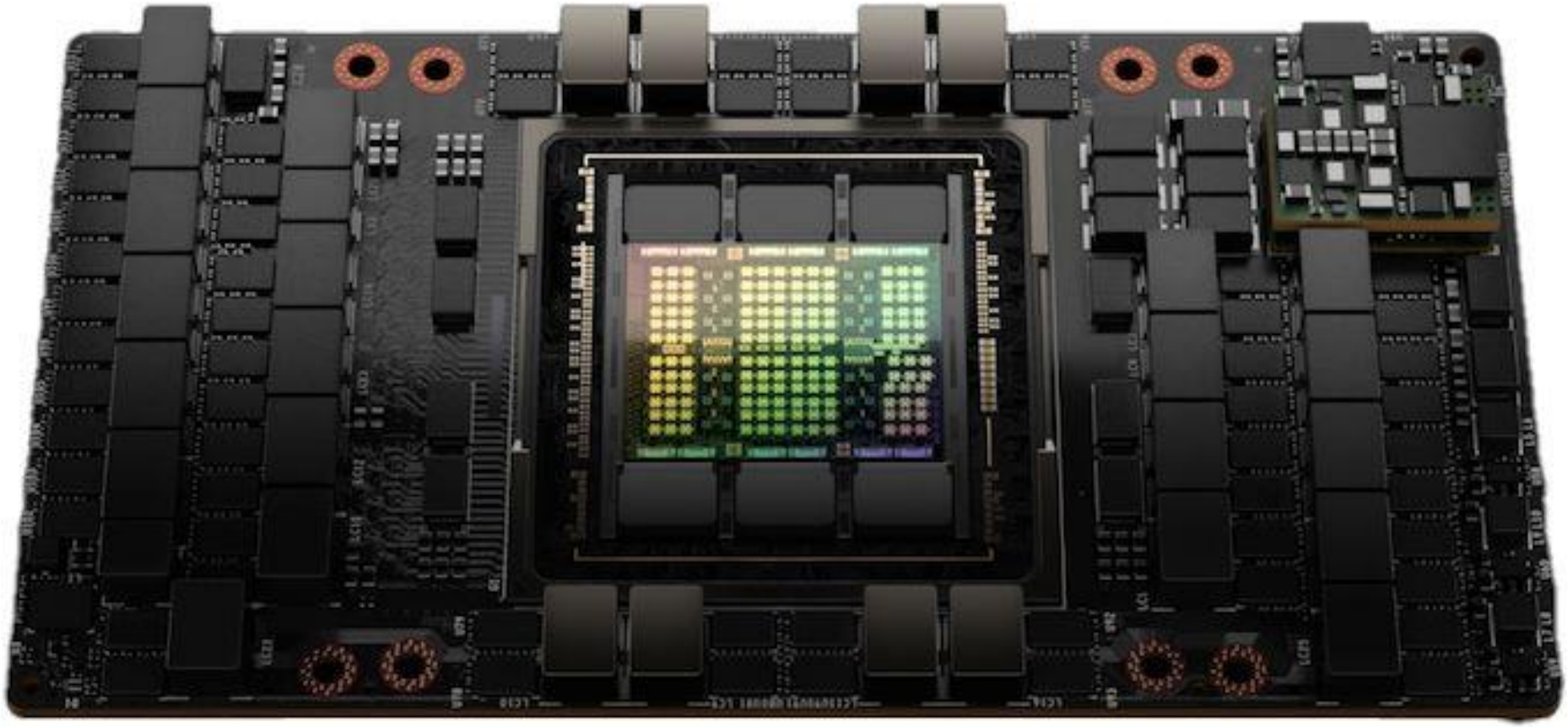
Different Platforms, Different Goals

- Tesla Dojo Chip & System

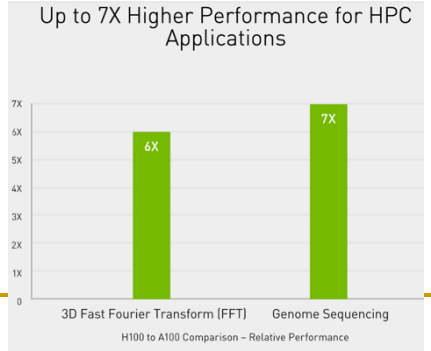


https://www.youtube.com/watch?v=j0z4FweCy4M&t=6340s

Different Platforms, Different Goals

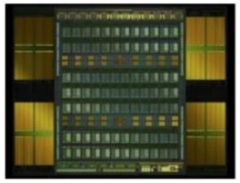


NVIDIA is claiming a **7x improvement** in dynamic programming algorithm (**DPX instructions**) performance on a single H100 versus naïve execution on an A100.

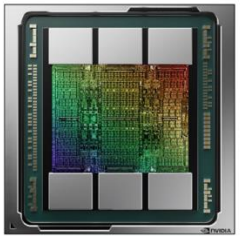


https://www.nvidia.com/en-us/data-center/h100/

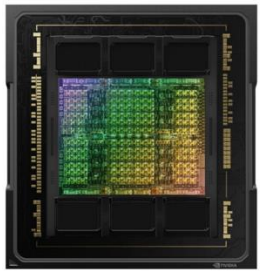
Evolution of Recent GPUs (I)



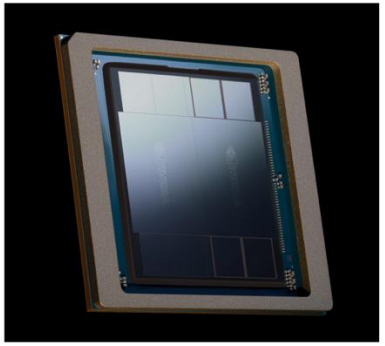
Volta
>21 billion transistors
815mm^2
TSMC 12nm FFN



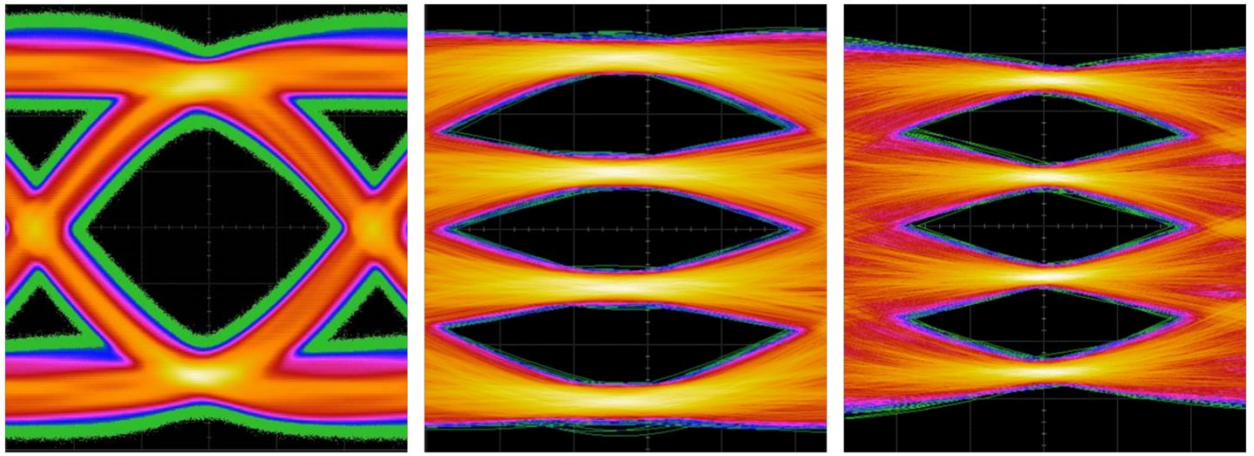
Ampere
>54 billion transistors
826 mm^2
TSMC N7



Hopper
>80 billion transistors
814 mm^2
TSMC 4N



Blackwell
>208 billion transistors
>1600 mm^2
TSMC 4NP

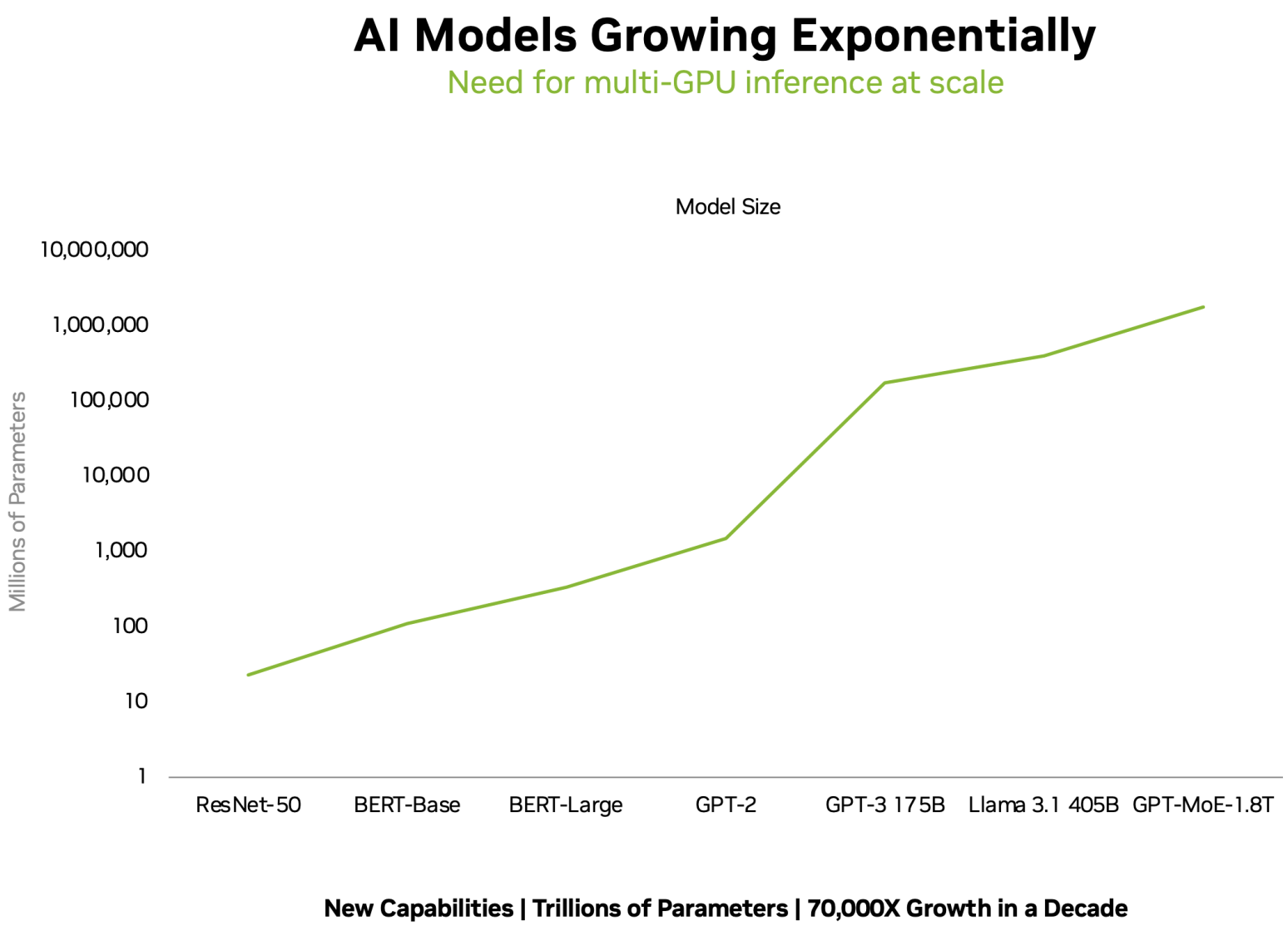


Ampere | NVLink3
12 NVLinks | 50GB/s each
x4@50Gbps-NRZ
600GB/s total

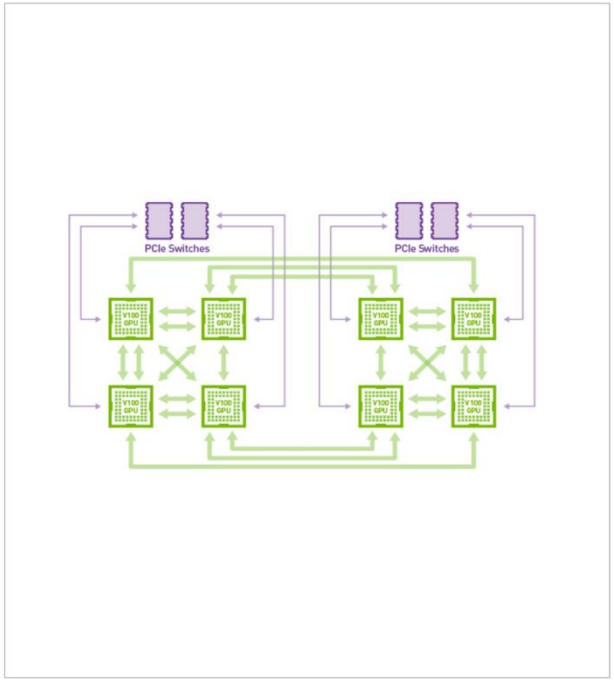
Hopper | NVLink4
18 NVLinks | 50GB/s each
x2@100Gbps-PAM4
900GB/s total

Blackwell | NVLink5
18 NVLinks | 100GB/s each
x2@200Gbps-PAM4
1800GB/s total

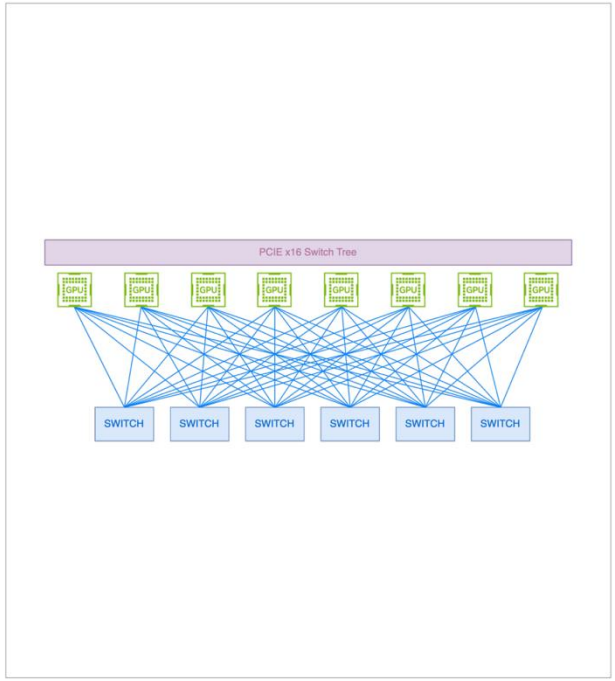
Multiple GPUs to Tackle Large Workloads



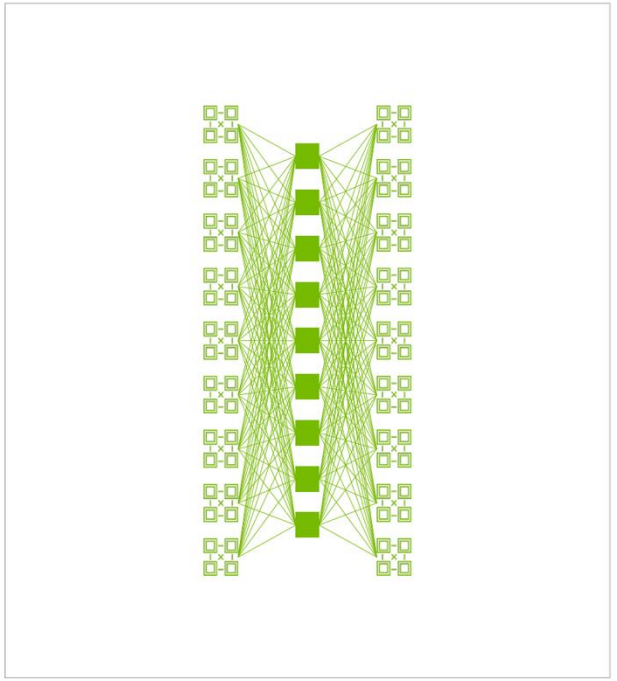
Evolution of Recent GPUs (II)



2016
Hybrid Cube Mesh NVLink technology

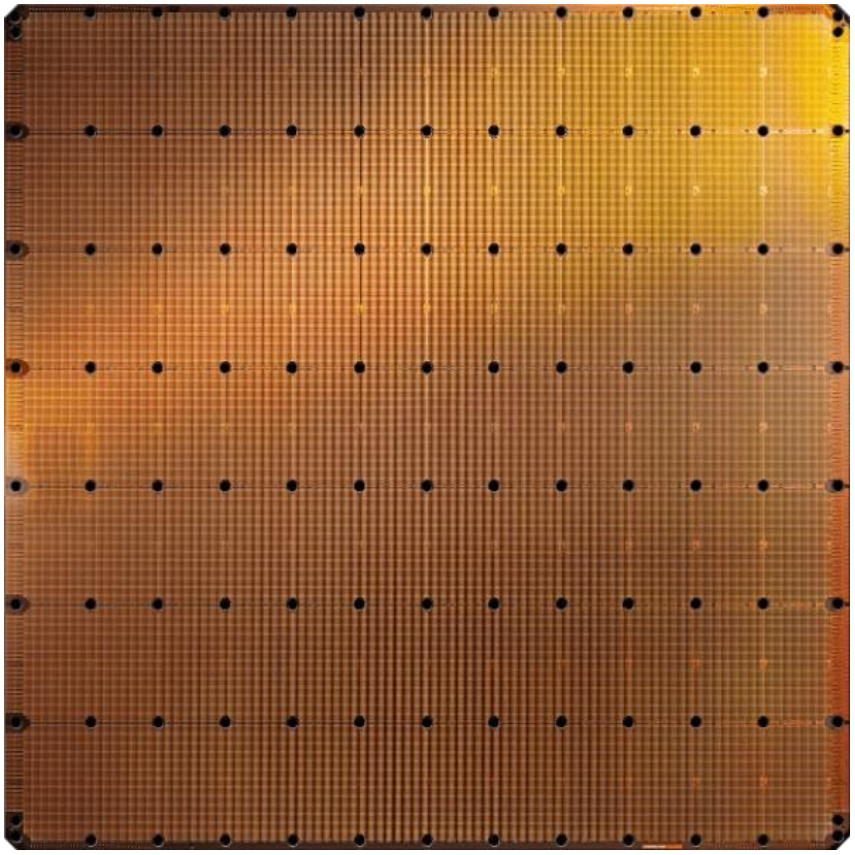


2022
3rd Gen NVLink Switch
All-to-all connection among NVLink domain of 8 GPU



2024
4th Gen NVLink Switch Chip
All-to-all connection among NVLink domain of 72 GPU

Different Platforms, Different Goals



- The largest ML accelerator chip (2021)
- 850,000 cores

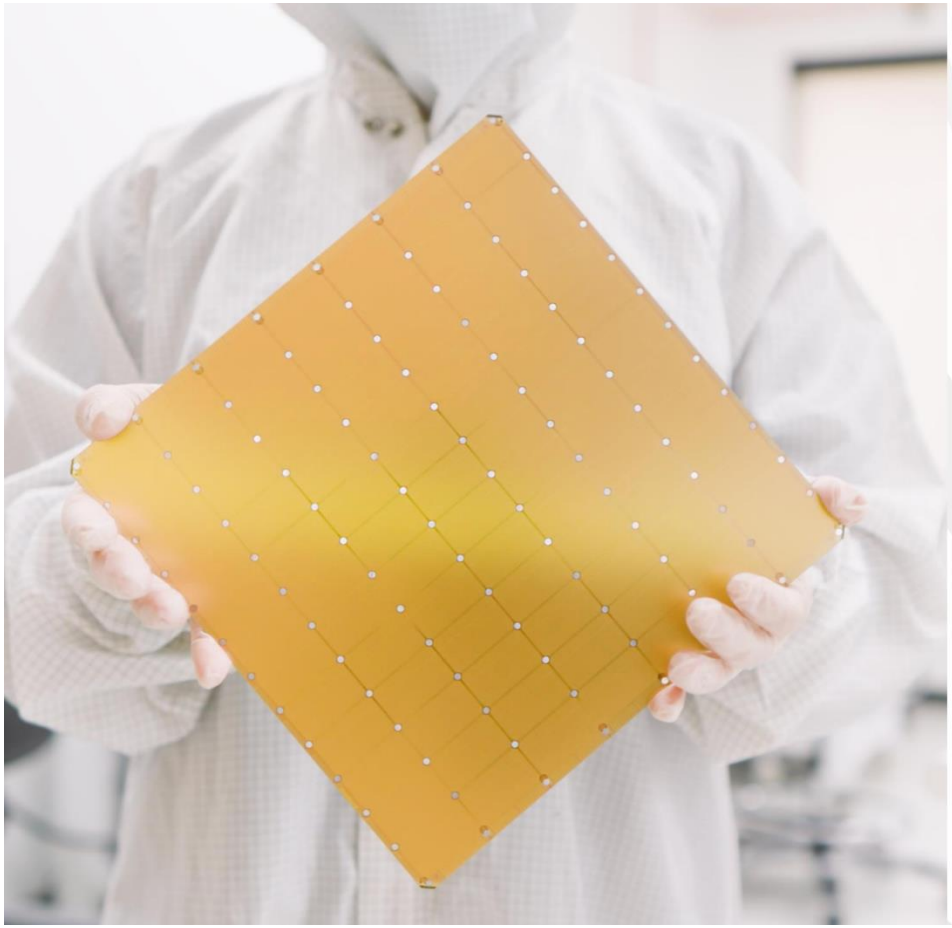
Cerebras WSE-2
2.6 Trillion transistors
46,225 mm^2

Largest GPU
54.2 Billion transistors
826 mm^2
NVIDIA Ampere GA100

https://www.anandtech.com/show/14758/hot-chips-31-live-blogs-cerebras-wafer-scale-deep-learning

https://www.cerebras.net/cerebras-wafer-scale-engine-why-we-need-big-chips-for-deep-learning/

Cerebras's Wafer Scale Engine-3 (2023)



Cerebras Wafer-Scale Engine

The largest chip ever produced

46,225 mm² silicon

4 trillion transistors

900,000 AI cores

125 Petaflops of AI compute

44 Gigabytes of on-chip memory

21 PByte/s memory bandwidth

214 Pbit/s fabric bandwidth

5nm TSMC process

Different Platforms, Different Goals

Mohammed Alser, Zülal Bingöl, Damla Senol Cali, Jeremie Kim, Saugata Ghose, Can Alkan, Onur Mutlu
"Accelerating Genome Analysis: A Primer on an Ongoing Journey" IEEE Micro, August 2020.

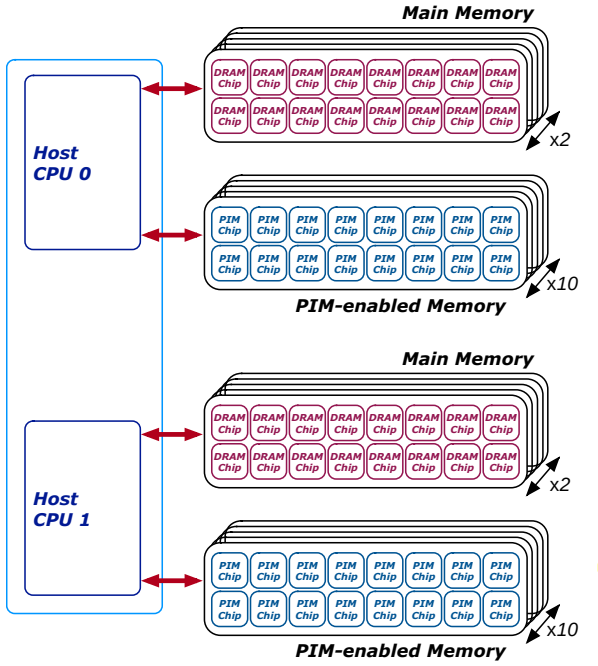
MinION from ONT

Accelerating Genome Analysis: A Primer on an Ongoing Journey
Sept.-Oct. 2020, pp. 65-75, vol. 40
DOI Bookmark: 10.1109/MM.2020.3013728

FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications
July-Aug. 2021, pp. 39-48, vol. 41
DOI Bookmark: 10.1109/MM.2021.3088396

SmidgION from ONT

Different Platforms, Different Goals



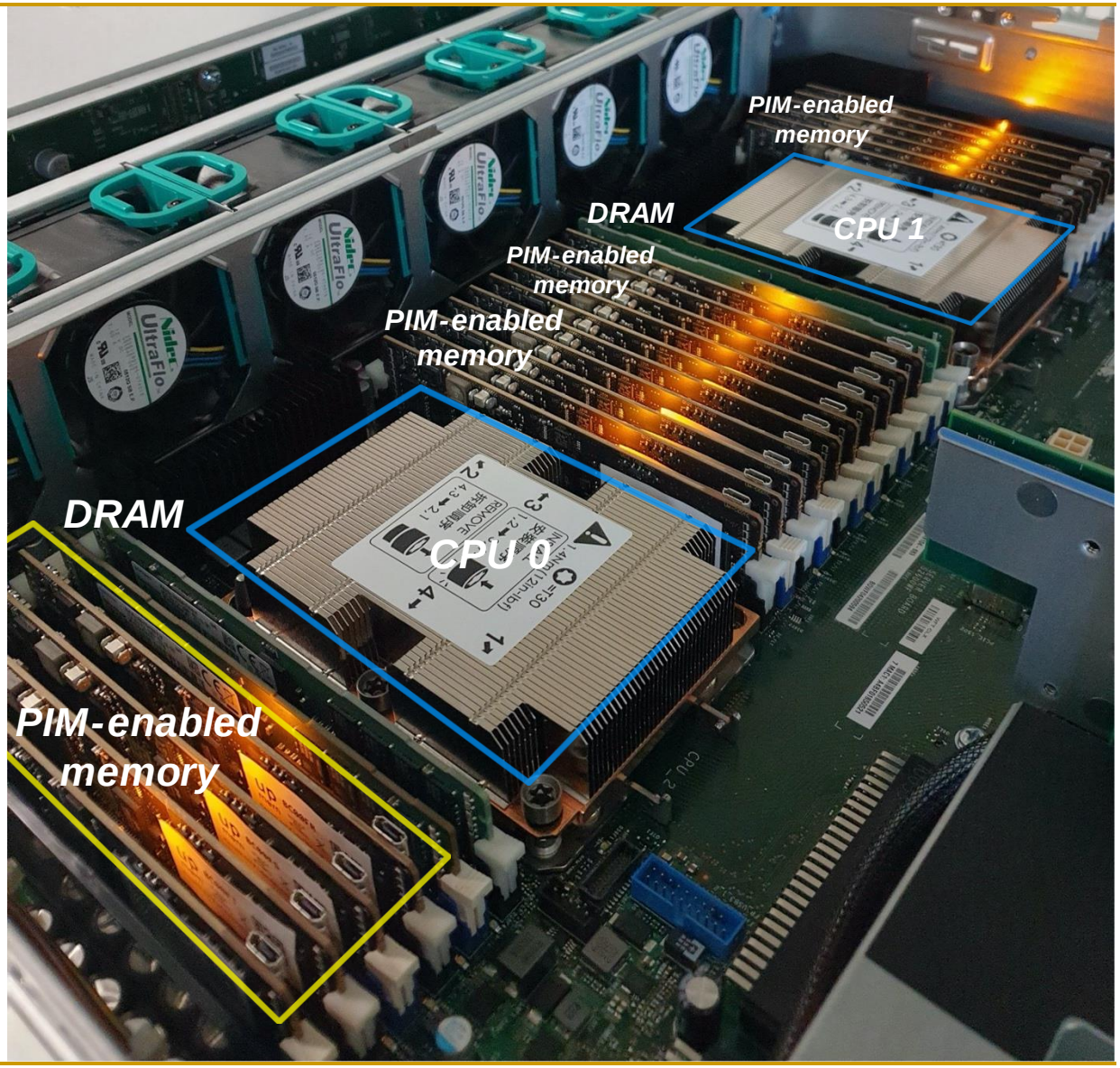
Benchmarking a New Paradigm: An Experimental Analysis of a Real Processing-in-Memory Architecture

JUAN GÓMEZ-LUNA, ETH Zürich, Switzerland
IZZAT EL HAJJ, American University of Beirut, Lebanon
IVAN FERNANDEZ, ETH Zürich, Switzerland and University of Malaga, Spain
CHRISTINA GIANNOULA, ETH Zürich, Switzerland and NTUA, Greece
GERALDO F. OLIVEIRA, ETH Zürich, Switzerland
ONUR MUTLU, ETH Zürich, Switzerland

Many modern workloads, such as neural networks, databases, and graph processing, are fundamentally memory-bound. For such workloads, the data movement between main memory and CPU cores imposes a significant overhead in terms of both latency and energy. A major reason is that this communication happens through a narrow bus with high latency and limited bandwidth, and the low data reuse in memory-bound workloads is insufficient to amortize the cost of main memory access. Fundamentally addressing this *data movement bottleneck* requires a paradigm where the memory system assumes an active role in computing by integrating processing capabilities. This paradigm is known as *processing-in-memory* (*PIM*).

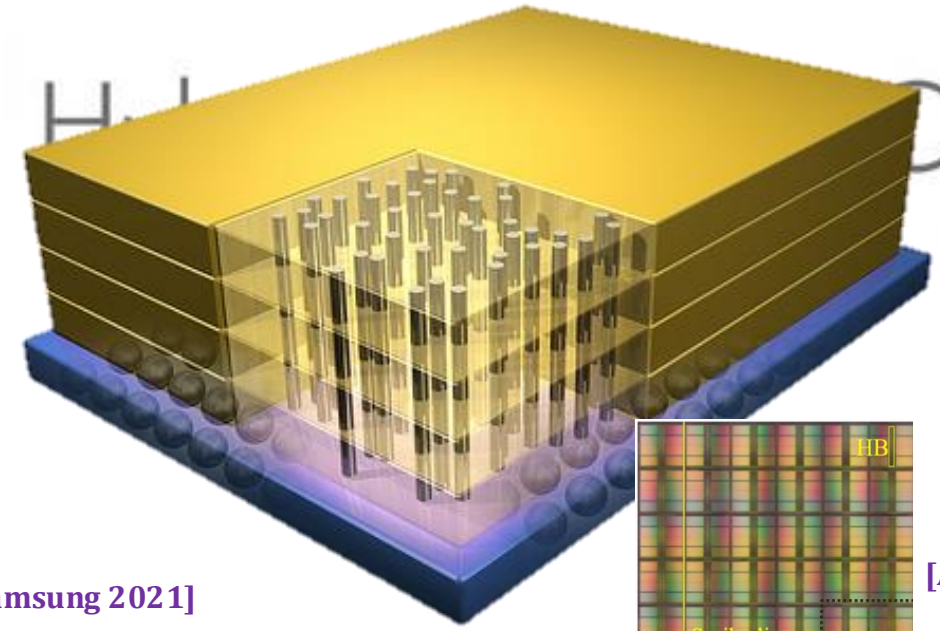
Recent research explores different forms of PIM architectures, motivated by the emergence of new 3D-stacked memory technologies that integrate memory with a logic layer where processing elements can be easily placed. Past works evaluate these architectures in simulation or, at best, with simplified hardware prototypes. In contrast, the UPMEM company has designed and manufactured the first publicly-available real-world PIM architecture. The UPMEM PIM architecture combines traditional DRAM memory arrays with general-purpose in-order cores, called *DRAM Processing Units* (*DPUs*), integrated in the same chip.

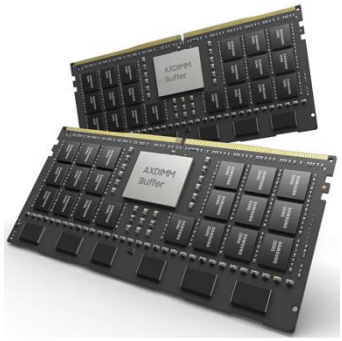
This paper provides the first comprehensive analysis of the first publicly-available real-world PIM architecture. We make two key contributions. First, we conduct an experimental characterization of the UPMEM-based PIM system using microbenchmarks to assess various architecture limits such as compute throughput and memory bandwidth, yielding new insights. Second, we present *PrIM* (*Processing-In-Memory benchmarks*), a benchmark suite of 16 workloads from different application domains (e.g., dense/sparse linear algebra, databases, data analytics, graph processing, neural networks, bioinformatics, image processing), which we identify as memory-bound. We evaluate the performance and scaling characteristics of PrIM benchmarks on the UPMEM PIM architecture, and compare their performance and energy consumption to their state-of-the-art CPU and GPU counterparts. Our extensive evaluation conducted on two real UPMEM-based PIM systems with 640 and 2,556 DPUs provides new insights about suitability of different workloads to the PIM system, programming recommendations for software designers, and suggestions and hints for hardware and architecture designers of future PIM systems.



https://arxiv.org/pdf/2105.03814.pdf

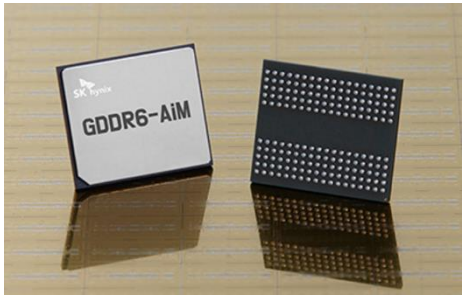
Processing-in-Memory Systems (2022)



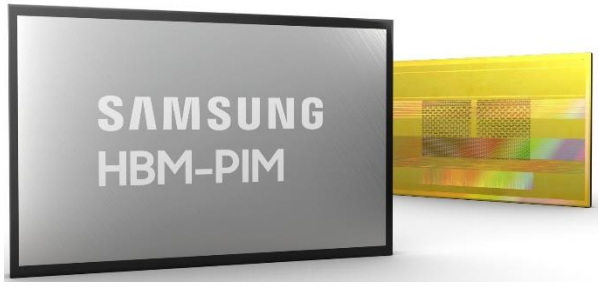


[Samsung 2021]

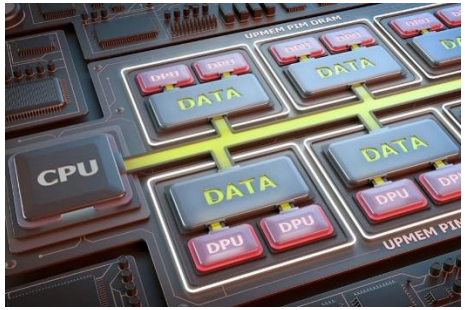
[Alibaba 2022]



[SK Hynix 2022]



[Samsung 2021]



[UPMEM 2019]

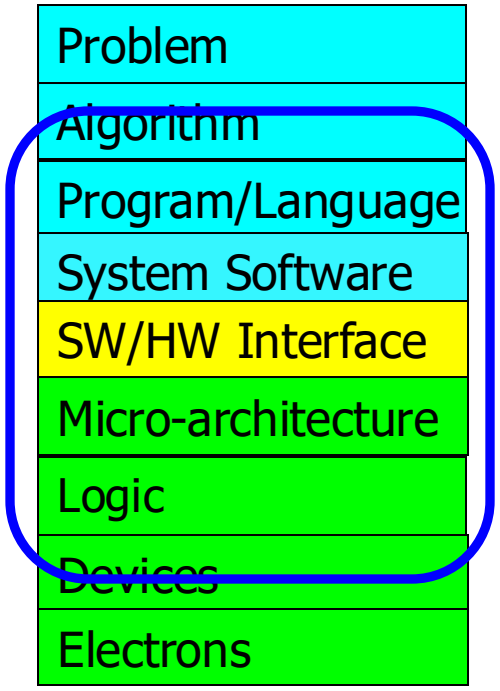
And, many other experimental chips and startups

Axiom

To achieve the highest energy efficiency and performance:

we must take the expanded view

of computer architecture



Co-design across the hierarchy: Algorithms to devices

Specialize as much as possible within the design goals

Why Study Computer Architecture?

What is Computer Architecture?

- 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.

Why Study Computer Architecture?

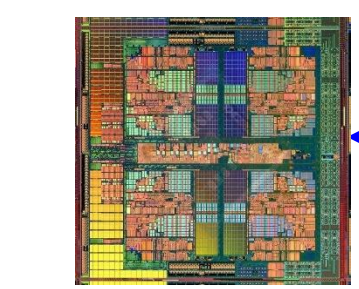
- Enable better systems: make computers faster, cheaper, smaller, more reliable, …
  - By exploiting advances and changes in underlying technology/circuits
- Enable new applications
  - Life-like 3D visualization 20 years ago? Virtual reality?
  - Self-driving cars?
  - Personalized genomics? Personalized medicine?
- Enable better solutions to problems
  - Software innovation is built on trends and changes in computer architecture
    - > 50% performance improvement per year has enabled this innovation
- Understand why computers work the way they do

Computer Architecture Today (I)

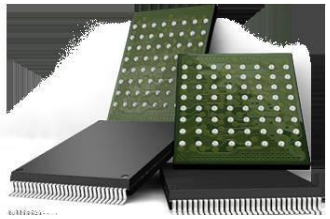
- Today is a very exciting time to study computer architecture
- Industry is in a large paradigm shift (to novel architectures) – many different potential system designs possible
- Many difficult problems *motivating* and *caused by* the shift
  - Huge hunger for data and new data-intensive applications
  - Power/energy/thermal constraints
  - Complexity of design
  - Difficulties in technology scaling
  - Memory bottleneck
  - Reliability problems
  - Programmability problems
  - Security and privacy issues
- No clear, definitive answers to these problems

Computer Architecture Today (II)

- Computing landscape is very different from 10-20 years ago
- Applications and technology both demand novel architectures

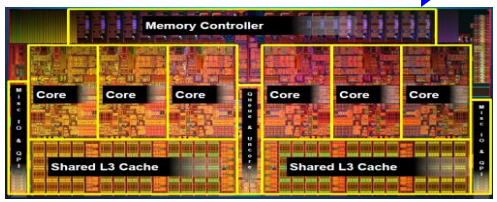


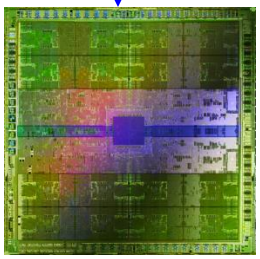
Hybrid Main Memory



Persistent Memory/Storage

Heterogeneous Processors and Accelerators

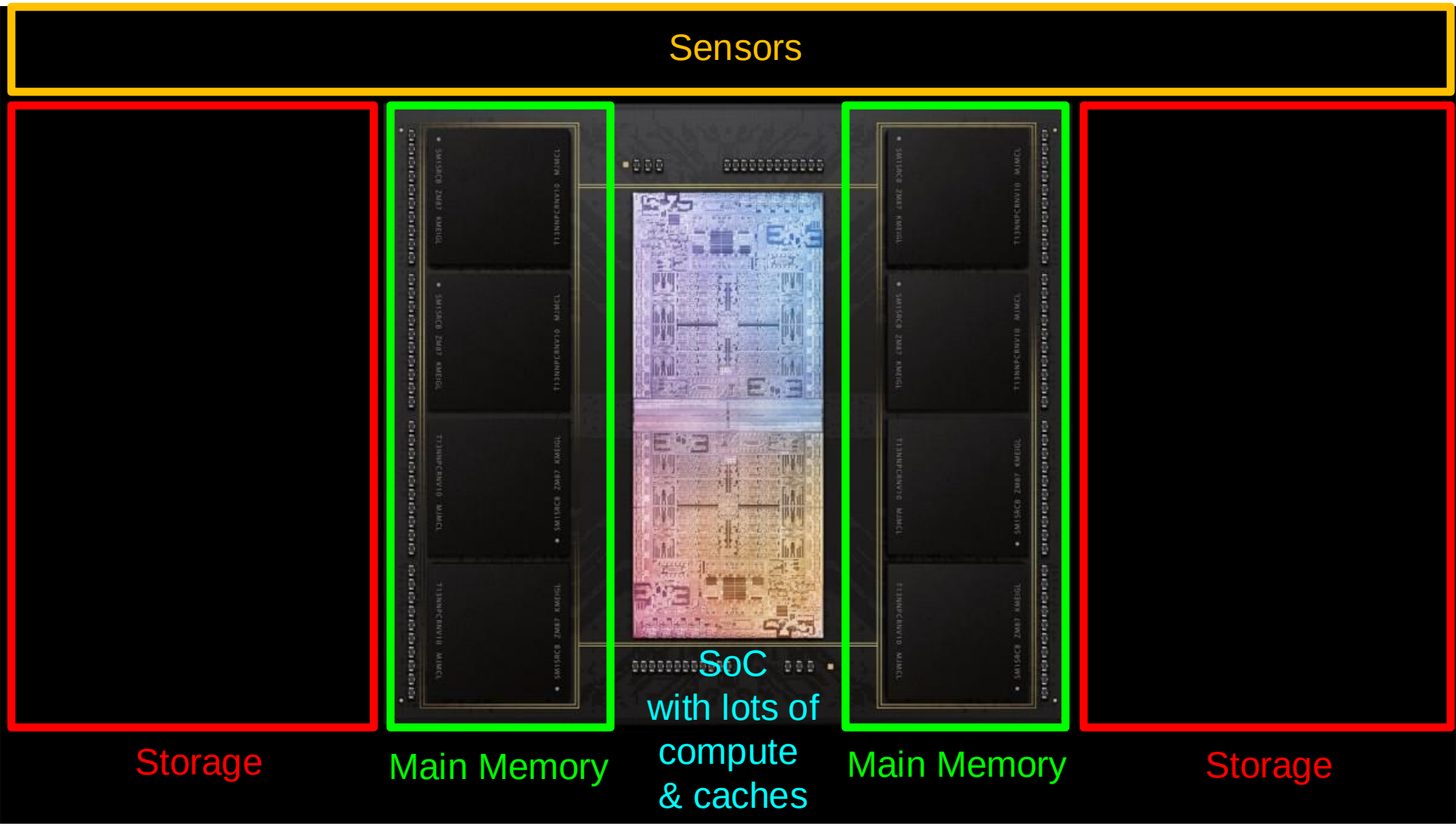




General Purpose GPUs

Every component and its interfaces, as well as entire system designs are being re-examined

An Example System in Your Pocket



Apple M1 Ultra System (2022)

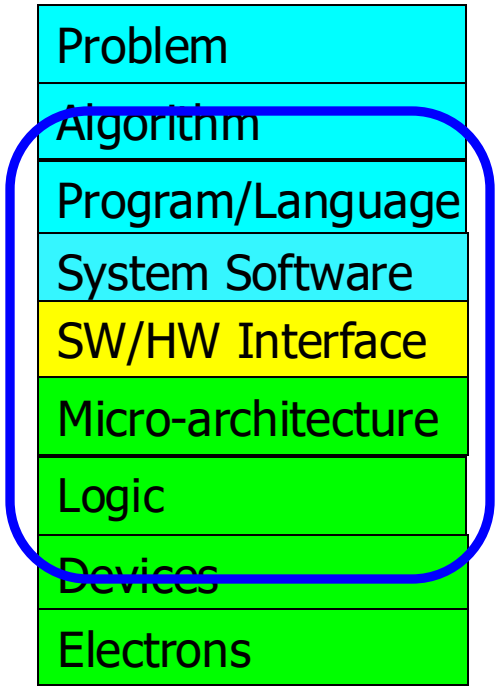
https://www.gsmarena.com/apple_announces_m1_ultra_with_20core_cpu_and_64core_gpu-news-53481.php

Axiom

To achieve the highest energy efficiency and performance:

we must take the expanded view

of computer architecture



Co-design across the hierarchy: Algorithms to devices

Specialize as much as possible within the design goals

Historical: Opportunities at the Bottom

There's Plenty of Room at the Bottom

From Wikipedia, the free encyclopedia

"**There's Plenty of Room at the Bottom: An Invitation to Enter a New Field of Physics**" was a lecture given by physicist Richard Feynman at the annual American Physical Society meeting at Caltech on December 29, 1959.[1] Feynman considered the possibility of direct manipulation of individual atoms as a more powerful form of synthetic chemistry than those used at the time. Although versions of the talk were reprinted in a few popular magazines, it went largely unnoticed and did not inspire the conceptual beginnings of the field. Beginning in the 1980s, nanotechnology advocates cited it to establish the scientific credibility of their work.

https://en.wikipedia.org/wiki/There%27s_Plenty_of_Room_at_the_Bottom

Historical: Opportunities at the Bottom (II)

There's Plenty of Room at the Bottom

From Wikipedia, the free encyclopedia

Feynman considered some ramifications of a general ability to manipulate matter on an atomic scale. He was particularly interested in the possibilities of denser computer circuitry, and microscopes that could see things much smaller than is possible with scanning electron microscopes. These ideas were later realized by the use of the scanning tunneling microscope, the atomic force microscope and other examples of scanning probe microscopy and storage systems such as Millipede, created by researchers at IBM.

Feynman also suggested that it should be possible, in principle, to make nanoscale machines that "arrange the atoms the way we want", and do chemical synthesis by mechanical manipulation.

He also presented the possibility of "swallowing the doctor", an idea that he credited in the essay to his friend and graduate student Albert Hibbs. This concept involved building a tiny, swallowable surgical robot.

https://en.wikipedia.org/wiki/There%27s_Plenty_of_Room_at_the_Bottom

Historical: Opportunities at the Top

REVIEW

There's plenty of room at the Top: What will drive computer performance after Moore's law?

Charles E. Leiserson[1], Neil C. Thompson[1,2,*], Joel S. Emer[1,3], Bradley C. Kuszmaul[1,†], Butler W. Lampson[1,4], ...
+ See all authors and affiliations

Science 05 Jun 2020:
Vol. 368, Issue 6495, eaam9744
DOI: 10.1126/science.aam9744

Much of the improvement in computer performance comes from decades of miniaturization of computer components, a trend that was foreseen by the Nobel Prize–winning physicist Richard Feynman in his 1959 address, "There's Plenty of Room at the Bottom," to the American Physical Society. In 1975, Intel founder Gordon Moore predicted the regularity of this miniaturization trend, now called Moore's law, which, until recently, doubled the number of transistors on computer chips every 2 years.

Unfortunately, semiconductor miniaturization is running out of steam as a viable way to grow computer performance—there isn't much more room at the "Bottom." If growth in computing power stalls, practically all industries will face challenges to their productivity. Nevertheless, opportunities for growth in computing performance will still be available, especially at the "Top" of the computing-technology stack: software, algorithms, and hardware architecture.

https://www.science.org/doi/10.1126/science.aam9744

Axiom, Revisited

There is plenty of room both at the top and at the bottom

but **much more so**

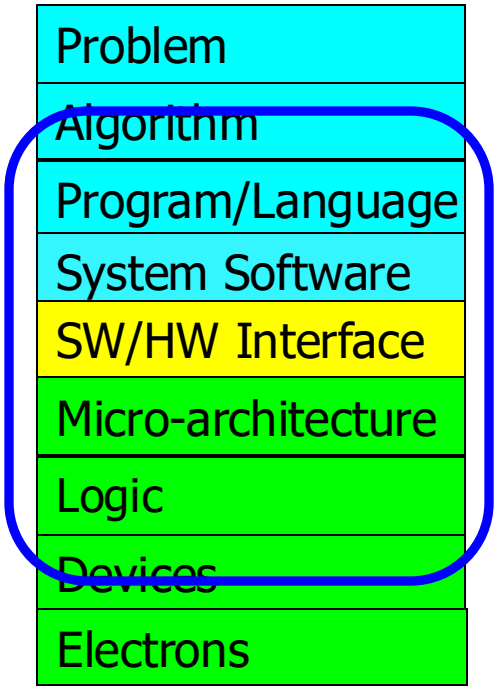
when you

communicate well between and optimize across

the top and the bottom

Hence the Expanded View

Computer Architecture (expanded view)



A Note on Hardware vs. Software

- This course might seem like it is only "Computer Hardware"
- However, you will be much more capable if you master both hardware and software (and the interface between them)
  - Can develop better software if you understand the hardware
  - Can design better hardware if you understand the software
  - Can design a better computing system if you understand both
- This course covers the HW/SW interface and microarchitecture
  - We will focus on tradeoffs and how they affect software
- Recall the example chips & platforms we surveyed

Course Info and Logistics

Brief Self Introduction

- Onur Mutlu
  - Full Professor @ ETH Zurich ITET (INFK), since Sept 2015
  - Strecker Professor @ Carnegie Mellon University ECE (CS), 2009-2016, 2016-…
  - Started the Comp Arch Research Group @ Microsoft Research, 2006-2009
  - Worked @ Google, VMware, Microsoft Research, Intel, AMD
  - PhD in Computer Engineering from University of Texas at Austin in 2006
  - BS in Computer Engineering & Psychology from University of Michigan in 2000
  - https://people.inf.ethz.ch/omutlu/ omutlu@gmail.com

- Research and Teaching in:
  - **Computer architecture, systems, hardware security, bioinformatics**
  - Memory and storage systems
  - Robust & dependable hardware systems: security, safety, predictability, reliability
  - Hardware/software cooperation
  - New computing paradigms; architectures with emerging technologies/devices
  - Architectures for bioinformatics, genomics, health, medicine, AI/ML
  - …

My Co-Instructor

- Mohammad Sadrosadati
  - Senior Researcher and Lecturer @ SAFARI Research Group, ETHZ
  - PhD in Computer Engineering from Sharif University of Technology in 2019
  - MS in Computer Engineering from Sharif University of Technology in 2014
  - BS in Computer Engineering from Sharif University of Technology in 2012
  - mohammad.sadrosadati@safari.ethz.ch

- Research & Teaching Areas
  - Computer Architecture
  - Memory/Storage Systems
  - Near-Data Processing
  - Heterogeneous System Architecture
  - Bioinformatics
  - Interconnection Networks

Head Teaching & Lab Assistants

- Dr. Konstantina Koliogeorgi
  - Head Teaching Assistant
  - Senior Researcher and Lecturer @ SAFARI
  - PhD, National Technical University of Athens, 2023

- Ataberk Olgun
  - Head Lab Assistant
  - PhD Student @ SAFARI

PhD/MSc Assistants

- (Other) Key Assistants and Guest Lecturers
  - Dr. Giray Yaglikci
  - Dr. Can Firtina
  - Geraldo De Oliveira Junior
  - Rahul Bera
  - Konstantinos Kanellopoulos
  - Nika Mansouri Ghiasi
  - Rakesh Nadig
  - Nisa Bostancı
  - İsmail Emir Yüksel
  - Haocong Luo
  - Andreas Kakolyris
  - Mayank Kabra
  - Jikun Wang
  - Susana Rebolledo Ruiz
  - Harshita Gupta

Student Assistants

- Maria Makeenkova
- Aaron Zeller
- Hafsa Sheikh Mohamud Ahmed
- Khushii Gupta
- Stanislav Drozhilkin
- Axel Schwarzenbach
- Cedric Koller
- Constantin Schweizer
- Henrik Pätzold
- Harsh Songara
- Ebruli Esin Doğan
- Julius Schneider
- Elena Lisa von Känel
- Joshua David Durrant
- Jonathan Soemer

Course Info: Lab Assistants (I)

- Tuesday 16-18
  - TBD
- Wednesday 16-18
  - TBD

Course Info: Lab Assistants (II)

- Friday 8-10
  - TBD
- Friday 10-12
  - TBD

If You Need Help

- Post your question on Moodle Q&A Forum
  - https://moodle-app2.let.ethz.ch/course/view.php?id=25002
  - We will create a forum on Moodle for each activity
  - **Preferred** for **technical** questions
- Write an e-mail to:
  - digitaltechnik@lists.inf.ethz.ch
  - The instructor and all assistants will receive this e-mail
- Come to office hours
  - We will provide office locations & Zoom links
  - TBD

Where to Get Up-to-date Course Info?

- Website:
  - **https://safari.ethz.ch/ddca/spring2025/**
  - Lecture slides and videos
  - Readings
  - Lab information
  - Course schedule, handouts, FAQs
  - Software
  - Any other useful information for the course
  - Check frequently for announcements and due dates
  - This is your single point of access to all resources
- Your ETH Email
- Lecturers and Teaching Assistants

Lecture and Lab Times and Policies

- Lectures:
  - Thursday and Fridays, 14:00-16:00
  - YouTube livestream playlist: https://www.youtube.com/playlist?list=PL5Q2soXY2Zi9Eo29LMgKVcaydS7V1zZW3
  - Zoom link provided via Moodle
  - Attendance is for your benefit and is therefore important
  - Some days, we may have guest lectures and exercise sessions
- Lab sessions:
  - See online
  - You should definitely attend the lab sessions
    - In-class evaluation (70%) and mandatory lab reports (30%)
  - Labs will start on March 4th
  - Lab information and handouts are here:
    - https://safari.ethz.ch/ddca/spring2025/doku.php?id=labs

Lab Organization (I)

- Groups
  - Choose your preferred group in Moodle
    - https://moodle-app2.let.ethz.ch/mod/choice/view.php?id=1189439
    - Due 01.03.2025 at 11:59pm
  - Choose your partner
    - https://moodle-app2.let.ethz.ch/mod/choicegroup/view.php?id=1189438
    - Due 25.02.2025 at 11:59pm
    - Fill this form ONLY if you have a partner
      - We will assign you a random partner if you do not fill the form

Lab Organization (II)

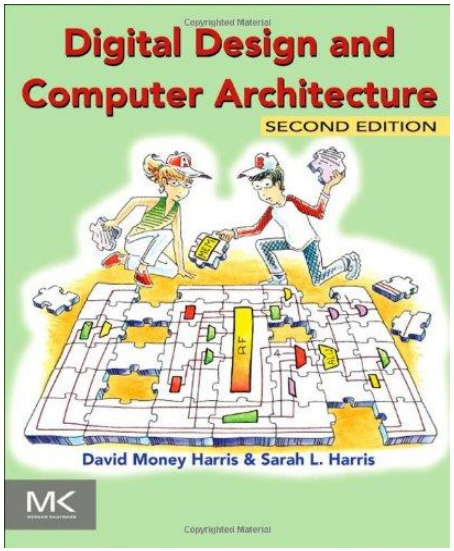
- Lab grades from previous years
  - https://moodle-app2.let.ethz.ch/mod/choice/view.php?id=1189441
  - Choose one of the options (due 01.03.2025 at 11:59pm):
    - 1) I will use my lab grades from previous years, and I won't do the labs this year
    - 2) I will use my lab grades from previous years, but I will do the labs this year
    - 3) I won't use my lab grades from previous years. I will do the labs this year

Final Exam

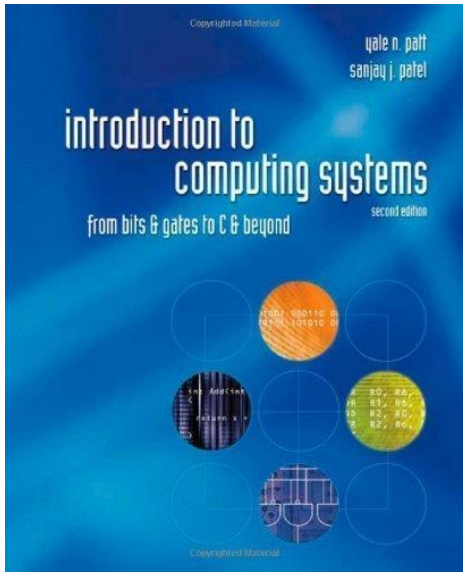
- 180-minute written exam
  - Find examination rules in Course Catalogue
    - https://www.vvz.ethz.ch/Vorlesungsverzeichnis/lerneinheit.view?lerneinheitId=188955&semkez=2025S&ansicht=LEHRVERANSTALTUNGEN&lang=en
  - Also: study the first pages of previous exams
    - https://safari.ethz.ch/digitaltechnik/spring2023/lib/exe/fetch.php?media=ddca-s23-en.pdf
  - Some exam questions are similar to questions in Optional HWs and Past Exams
    - Optional HWs are not graded, but highly recommended to solve
    - Solving past exams could also be useful

Reading Assignments for This/Next Week

- Chapters 1-2 in Harris & Harris



- Chapters 1,2,3 in Patt and Patel



- Supplementary Lecture Slides on Binary Numbers

Reading Assignments for This/Next Week

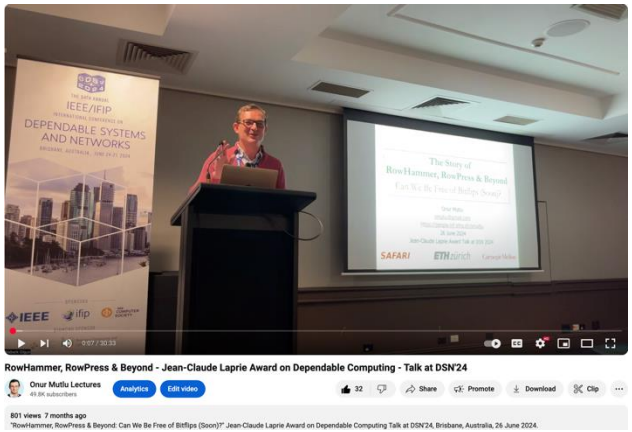
- This week
  - Introduction
    - P&P Chapters 1 & 2 + H&H Chapter 1
  - Combinational Logic
    - P&P Chapter 3 until 3.3 + H&H Chapter 2

- Next week
  - Hardware Description Languages and Verilog
    - H&H Chapter 4 until 4.3 and 4.5
  - Sequential Logic
    - P&P Chapter 3.4 until end + H&H Chapter 3 in full

- Within 2-3 weeks, we will be done with
  - **P&P Chapters 1-3 + H&H Chapters 1-4**

Extra Credit Assignment: Talk Analysis

- The Story of RowHammer, RowPress & Beyond
- **Watch and analyze this short lecture (30 minutes)**
  - https://www.youtube.com/watch?v=U1EcqXlclKU (June 2024)



- **Assignment – for 1% extra credit**
  - **Write a good 1-page individualized summary**
    - What are your key takeaways? What did you learn?
    - What surprised you about the content presented? What excited you?
    - What do you think solutions should be like?
    - Submit your summary to Moodle – deadline March 21

Future Lectures and Assignments

- You can also anticipate (and plan for) future lectures and assignments based on the Spring 2023 schedule:
  - https://safari.ethz.ch/digitaltechnik/spring2023/doku.php?id=schedule
  - https://www.youtube.com/playlist?list=PL5Q2soXY2Zi-EImKxYYY1SZuGiOAOBKaf

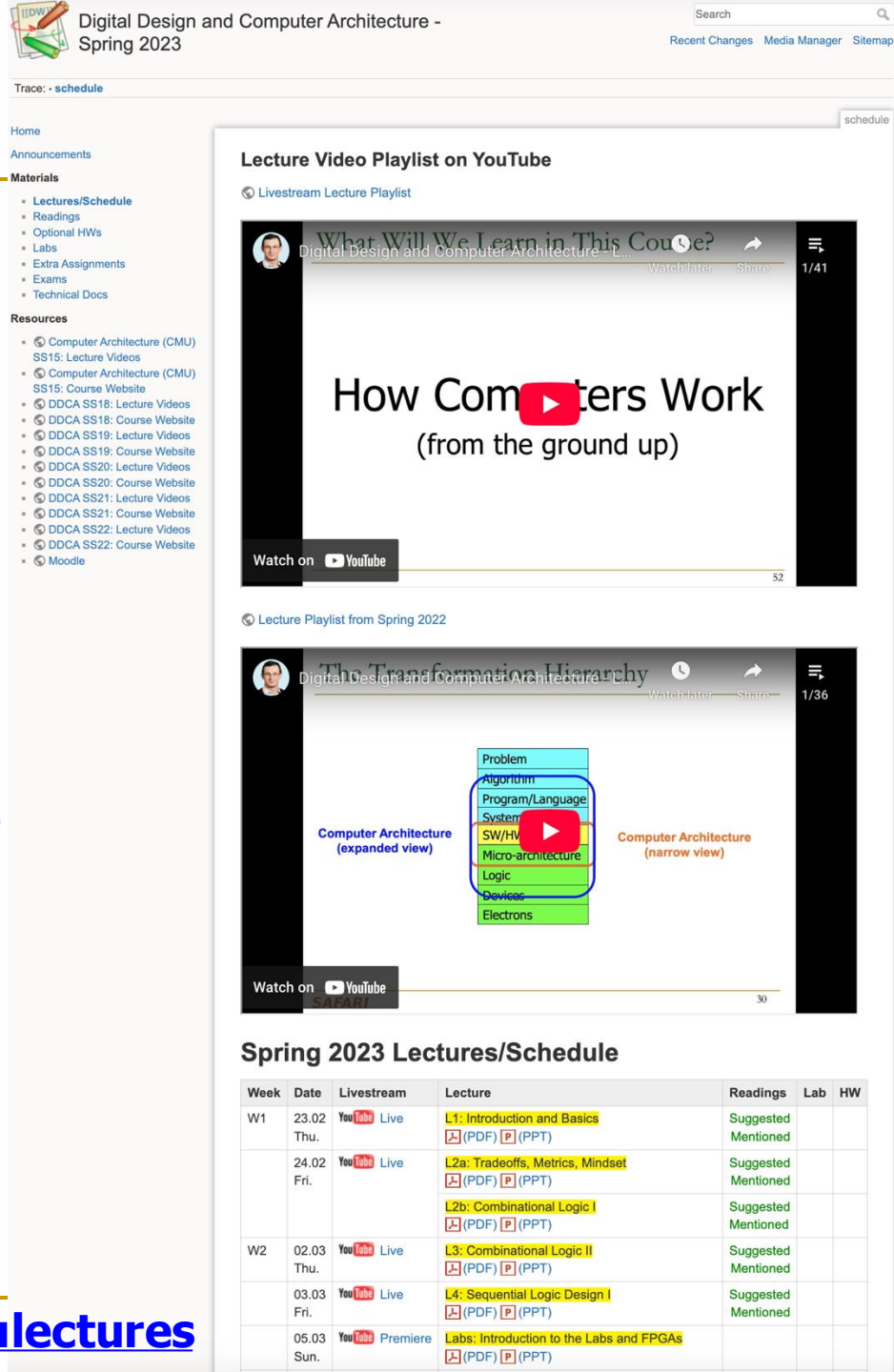
- An example of "Last Time Prediction"
  - Speculative Execution
    - The concept of doing something before knowing it is needed
  - A key concept we will cover in the design of microprocessors

DDCA (Spring 2023)

- **Spring 2023 Edition:**
  - https://safari.ethz.ch/digitaltechnik/spring2023/doku.php?id=schedule
- **Spring 2022 Edition:**
  - https://safari.ethz.ch/digitaltechnik/spring2022/doku.php?id=schedule
- **Spring 2021 Edition:**
  - https://safari.ethz.ch/digitaltechnik/spring2021/doku.php?id=schedule

- **Youtube Livestream (Spring 2023):**
  - https://www.youtube.com/watch?v=VcKjvwD930o&list=PL5Q2soXY2Zi-EImKxYYY1SZuGiOAOBKaf
- **Youtube Livestream (Spring 2022):**
  - https://www.youtube.com/watch?v=cpXdE3HwvK0&list=PL5Q2soXY2Zi97Ya5DEUpMpO2bbAoaG7c6
- **Youtube Livestream (Spring 2021):**
  - https://www.youtube.com/watch?v=LbC0EZY8yw4&list=PL5Q2soXY2Zi_uej3aY39YB5pfW4SJ7LlN

https://www.youtube.com/onurmutlulectures



Fundamental Concepts

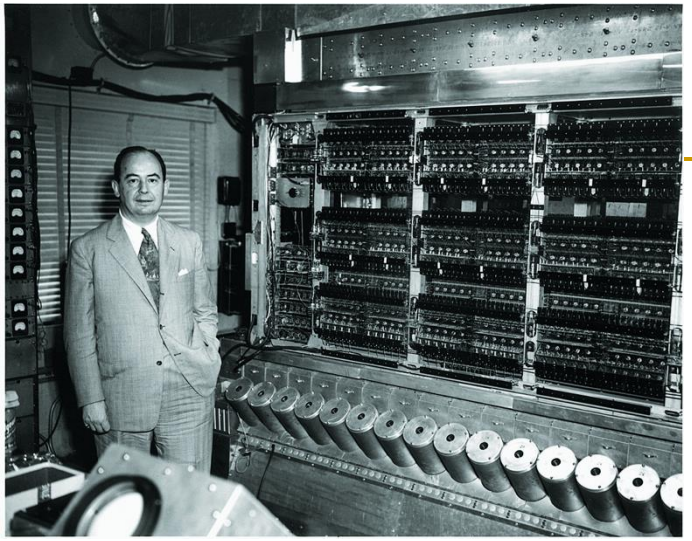
First …

- Let's understand the fundamentals…

- You can change the world only if you understand it well enough…
  - Especially the basics (fundamentals)
  - Past and present dominant paradigms
  - And, their advantages and shortcomings – tradeoffs
  - And, what remains fundamental across generations
  - And, what techniques you can use and develop to solve problems

What is A Computer?

- Three key components
- Computation
- Communication
- Storage/memory



Burks, Goldstein, von Neumann, "Preliminary discussion of the logical design of an electronic computing instrument," 1946.

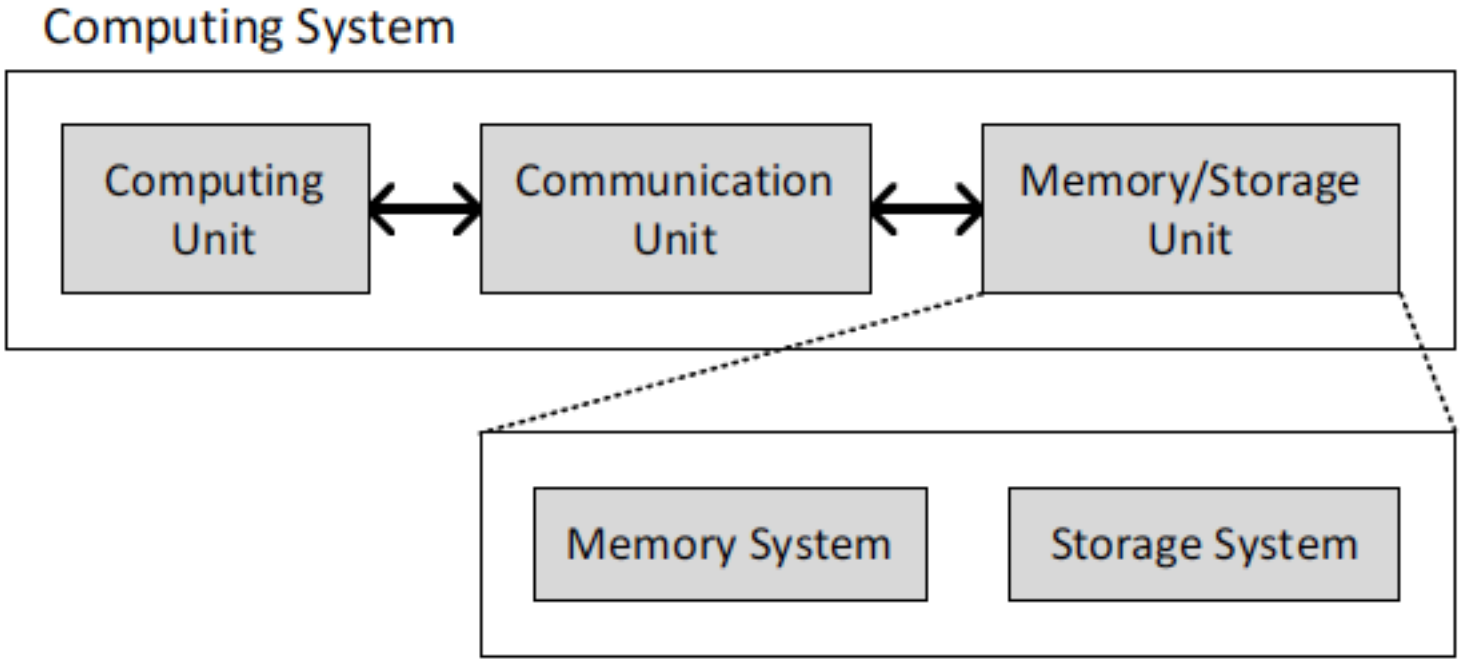
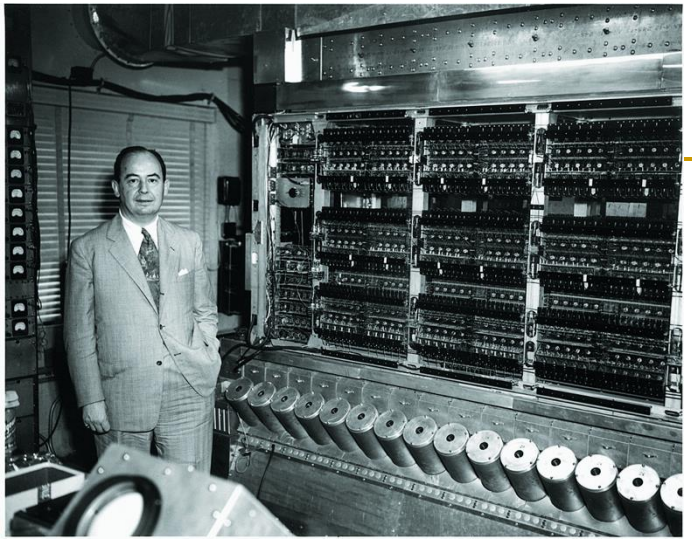


Image source: https://lbsitbytes2010.wordpress.com/2013/03/29/john-von-neumann-roll-no-15/

What is A Computer?

- Three key components
- Computation
- Communication
- Storage/memory



Burks, Goldstein, von Neumann, "Preliminary discussion of the logical design of an electronic computing instrument," 1946.

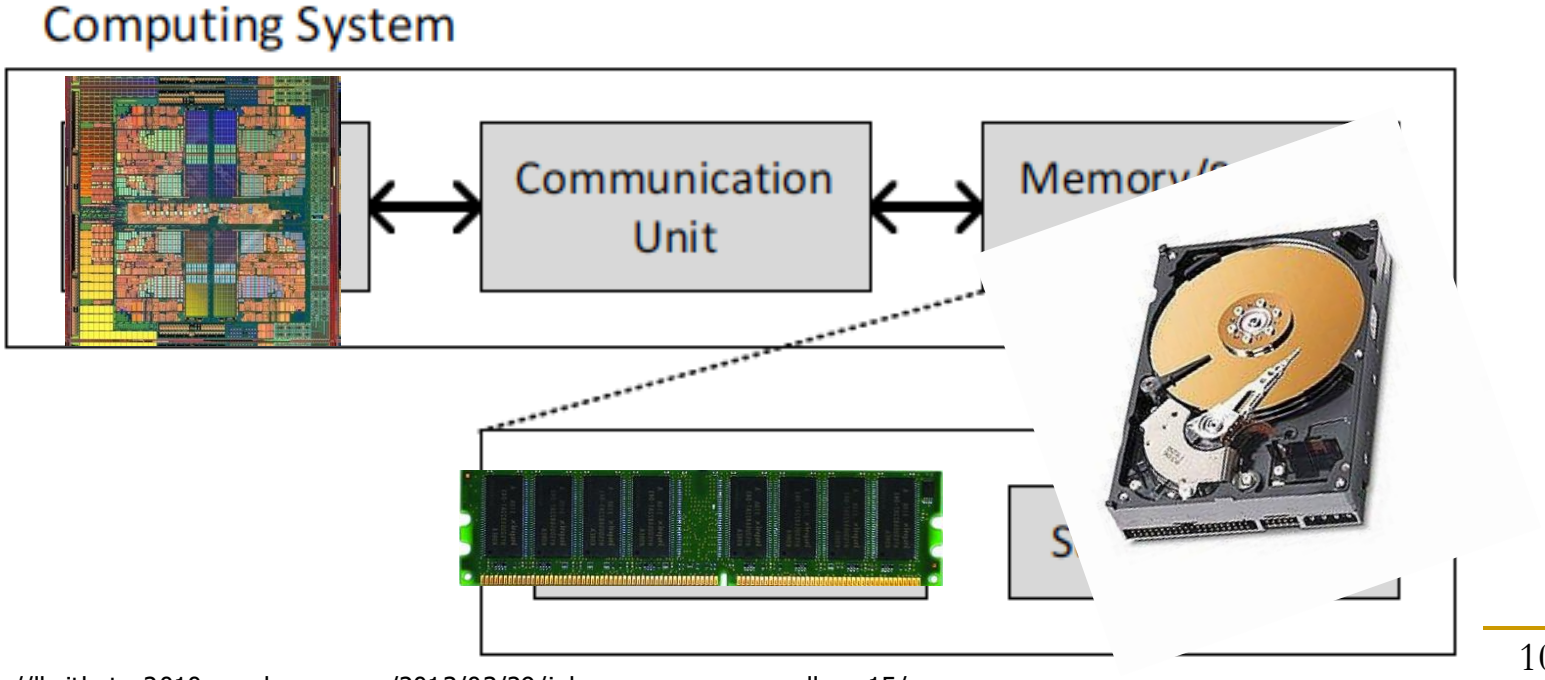
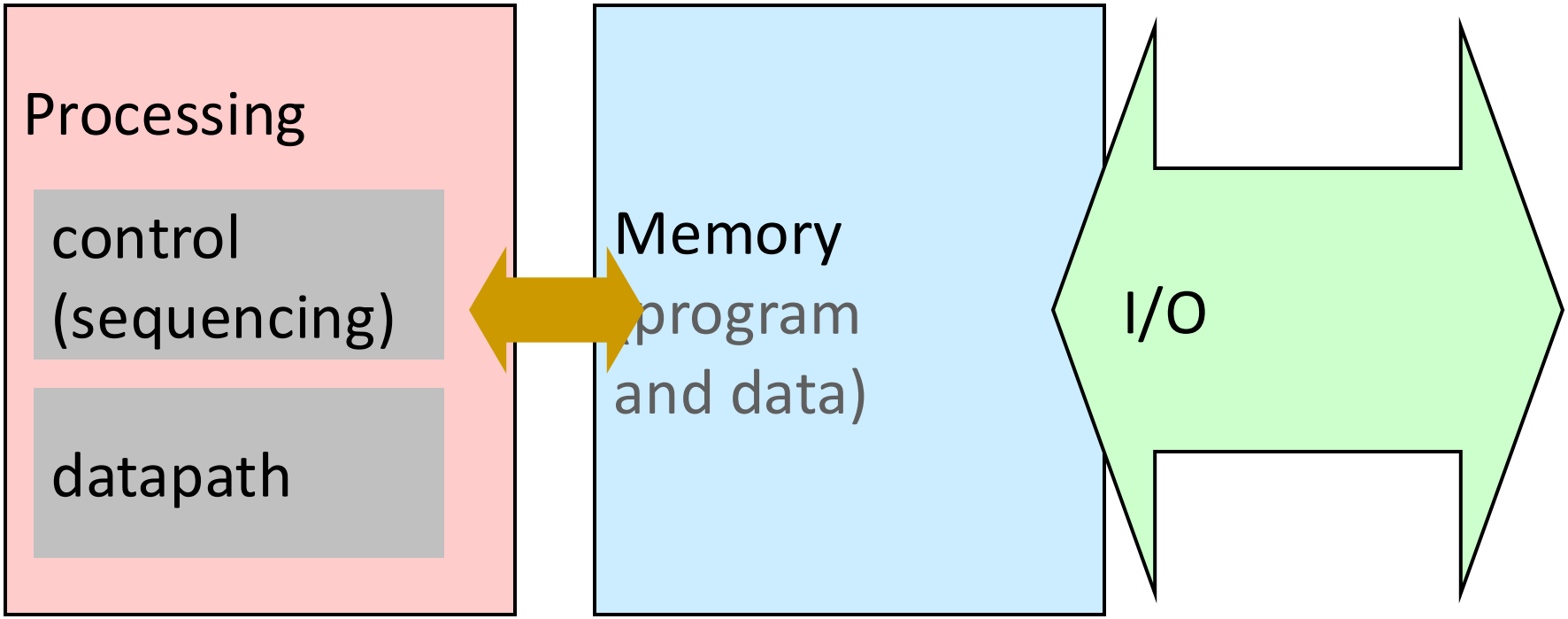


Image source: https://lbsitbytes2010.wordpress.com/2013/03/29/john-von-neumann-roll-no-15/

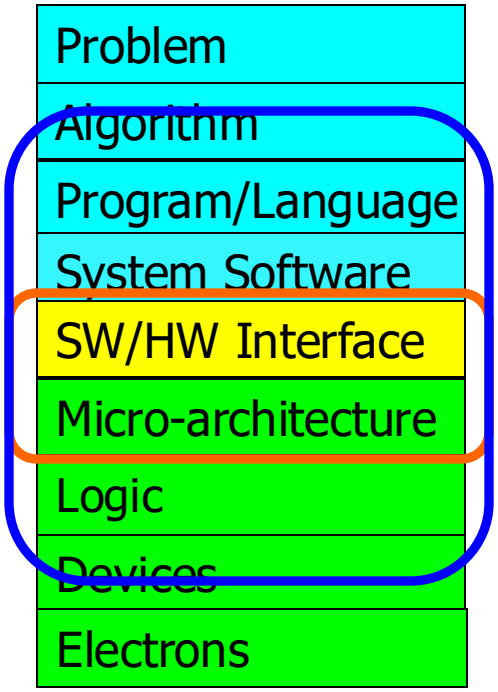
What is A Computer?

- We will cover all three components



Recall: The Transformation Hierarchy

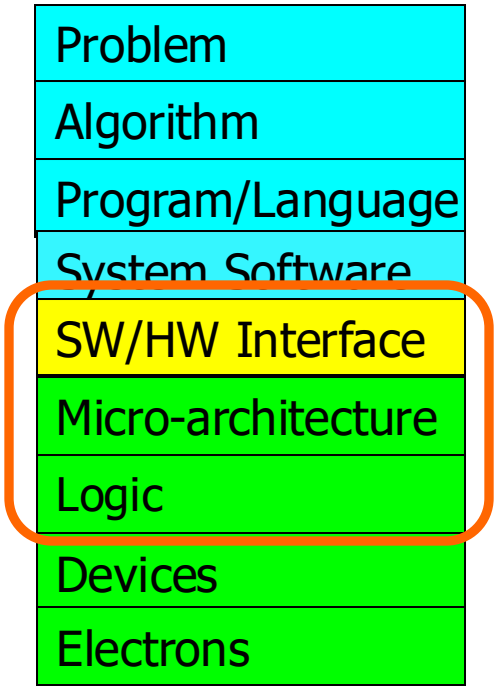
Computer Architecture (expanded view)



Computer Architecture (narrow view)

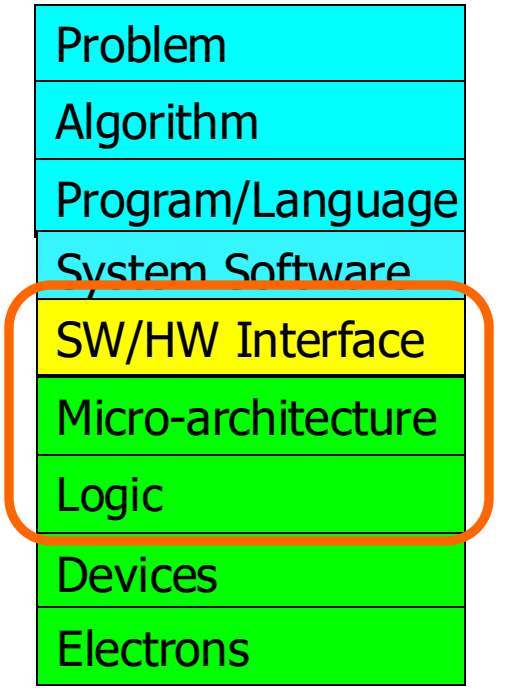
What We Will Cover (I)

- Combinational Logic Design
- Hardware Description Languages (Verilog)
- Sequential Logic Design
- Timing and Verification
- ISA (MIPS and LC3b as examples)
- Assembly Programming



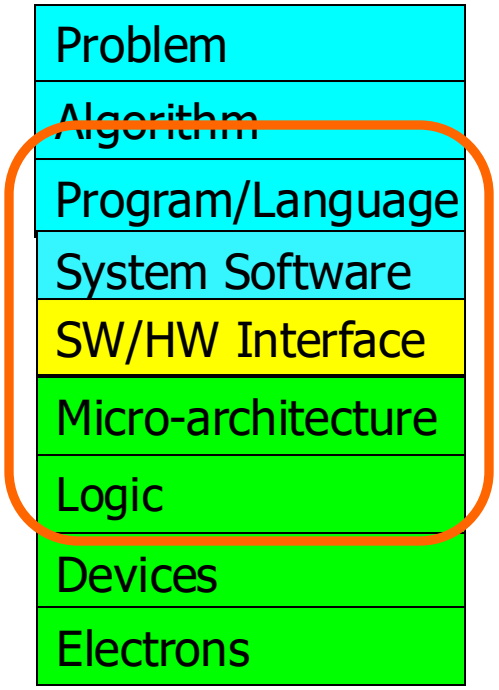
What We Will Cover (II)

- Microarchitecture Fundamentals
- Single-cycle Microarchitectures
- Multi-cycle and Microprogrammed Microarchitectures
- Pipelining
- Issues in Pipelining: Dependence Handling, State Maintenance and Recovery, …
- Branch Prediction
- Out-of-Order Execution
- Superscalar Execution
- Other Paradigms: Dataflow, VLIW, Systolic, SIMD/GPUs, …



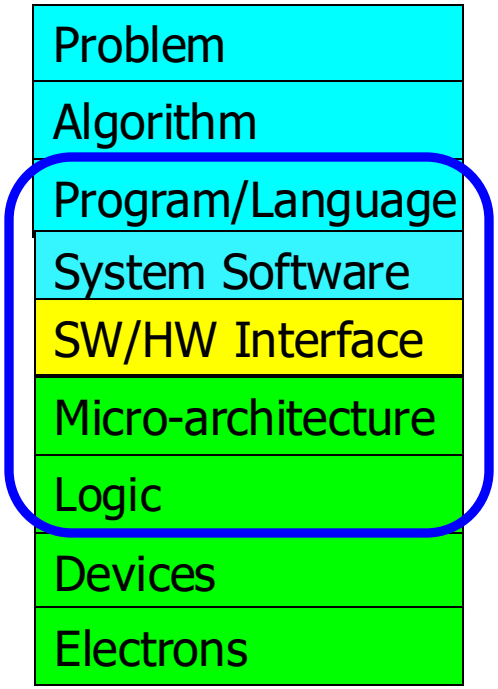
What We Will Cover (III)

- Memory Technology and Organization
- Memory Hierarchy
- Caches
- Multi-Core Caches
- Prefetching
- Virtual Memory



Processing Paradigms We Will Cover

- Pipelining
- Out-of-order execution
- Dataflow (at the ISA level)
- Superscalar Execution
- VLIW
- Decoupled Access-Execute
- Systolic Arrays
- SIMD Processing (Vector & Array)
- GPUs



Combinational Logic Circuits and Design

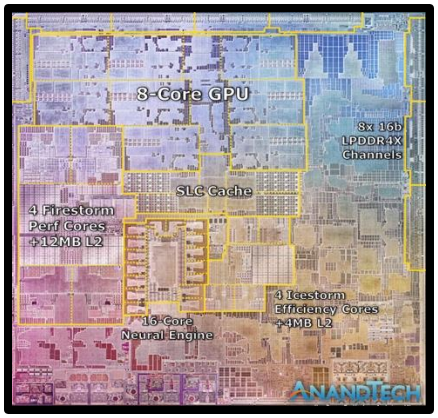
What Will We Learn Today & Tomorrow?

- Basic building blocks of modern computers
  - Transistors
  - Logic gates
- Boolean algebra
- Combinational logic circuits
- How to use Boolean algebra to represent combinational circuits
- Minimizing logic circuits (if time permits)

General Purpose vs. Special Purpose Systems

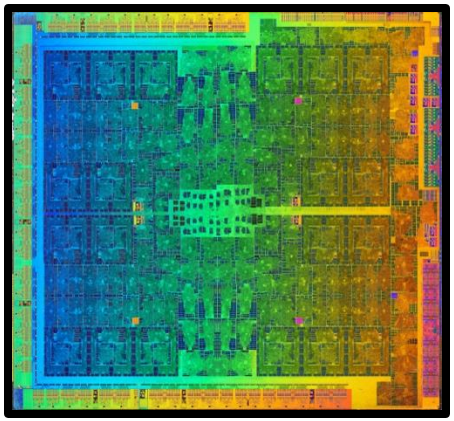
General Purpose

CPUs



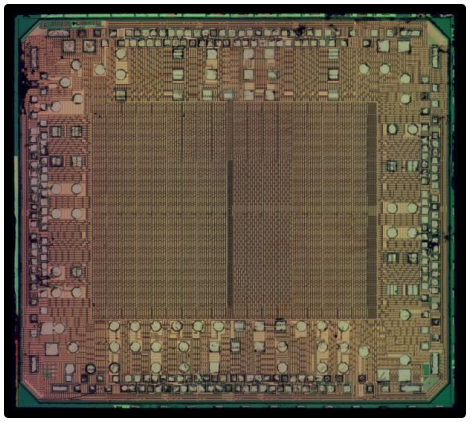
Apple M1

GPUs



Nvidia GTX 1070

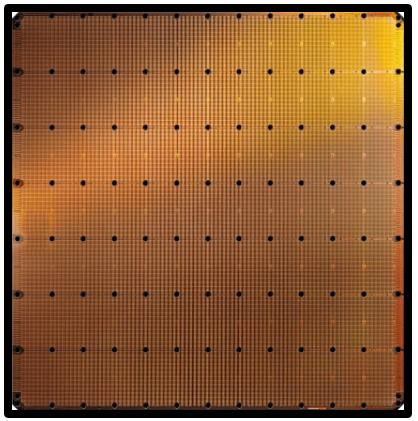
FPGAs



Xilinx Spartan

Special Purpose

ASICs



Cerebras WSE-2

Flexible: Can execute any program

Easy to program & use

Not the best performance & efficiency

Efficient & High performance

(Usually) Difficult to program & use

Inflexible: Limited set of programs

General Purpose vs. Special Purpose Systems

General Purpose

Special Purpose

Flexible: Can work with any bolt
Easy to use
Not the best fit, results or efficiency

Efficient & High performance
(Usually) Difficult to use
Inflexible: Works only for fitting bolts

https://www.ubuy.vn/en/product/2MBUW8M-crescent-8-adjustable-wrench-carded-ac28vs
https://capritools.com/shop/bolt-extractor-wrench-set-metric-8-19-mm/

https://www.vigor-equipment.com/en/combination-wrench-4971

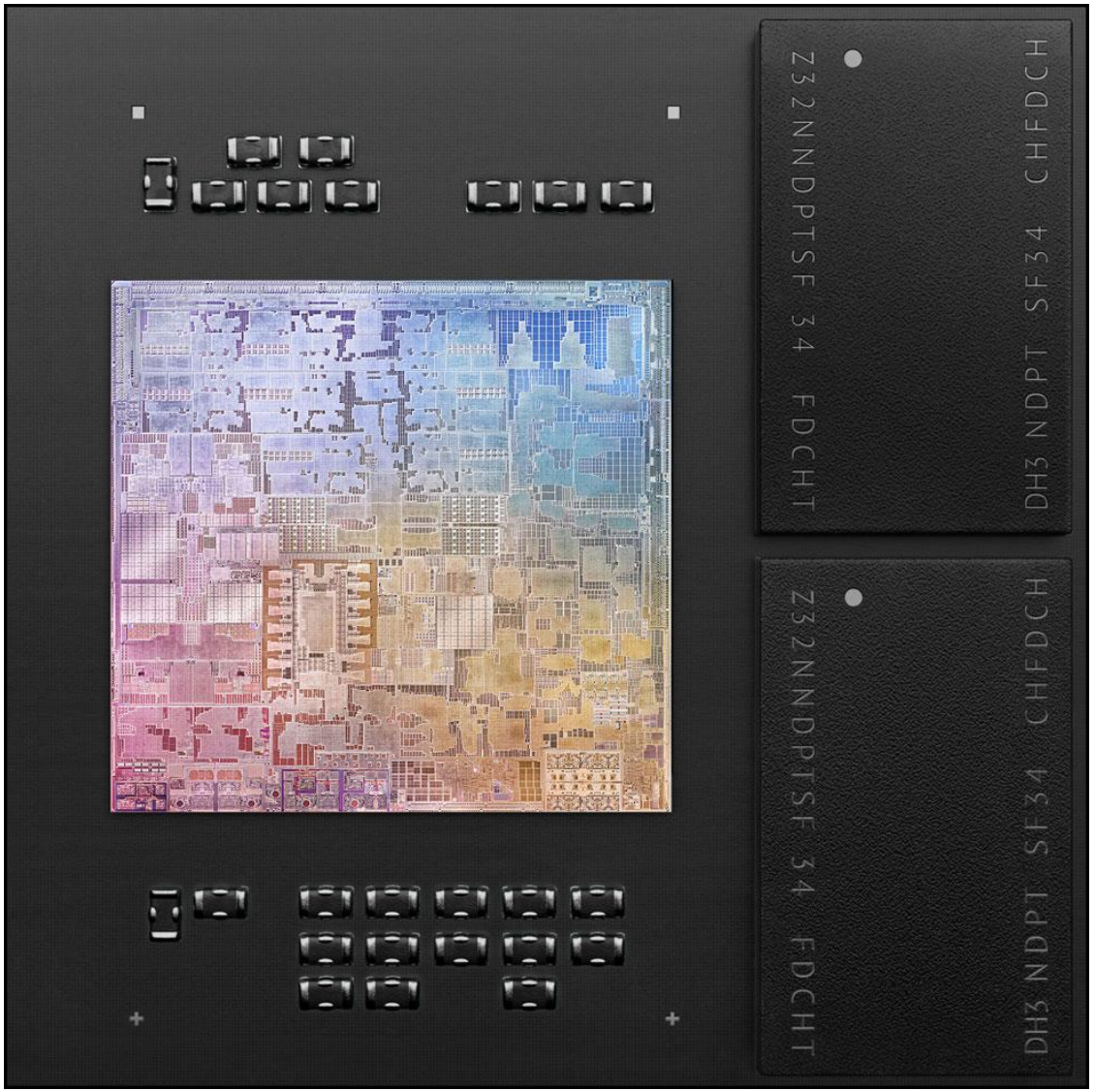
Modern General-Purpose Microprocessors

5-nanometer process

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

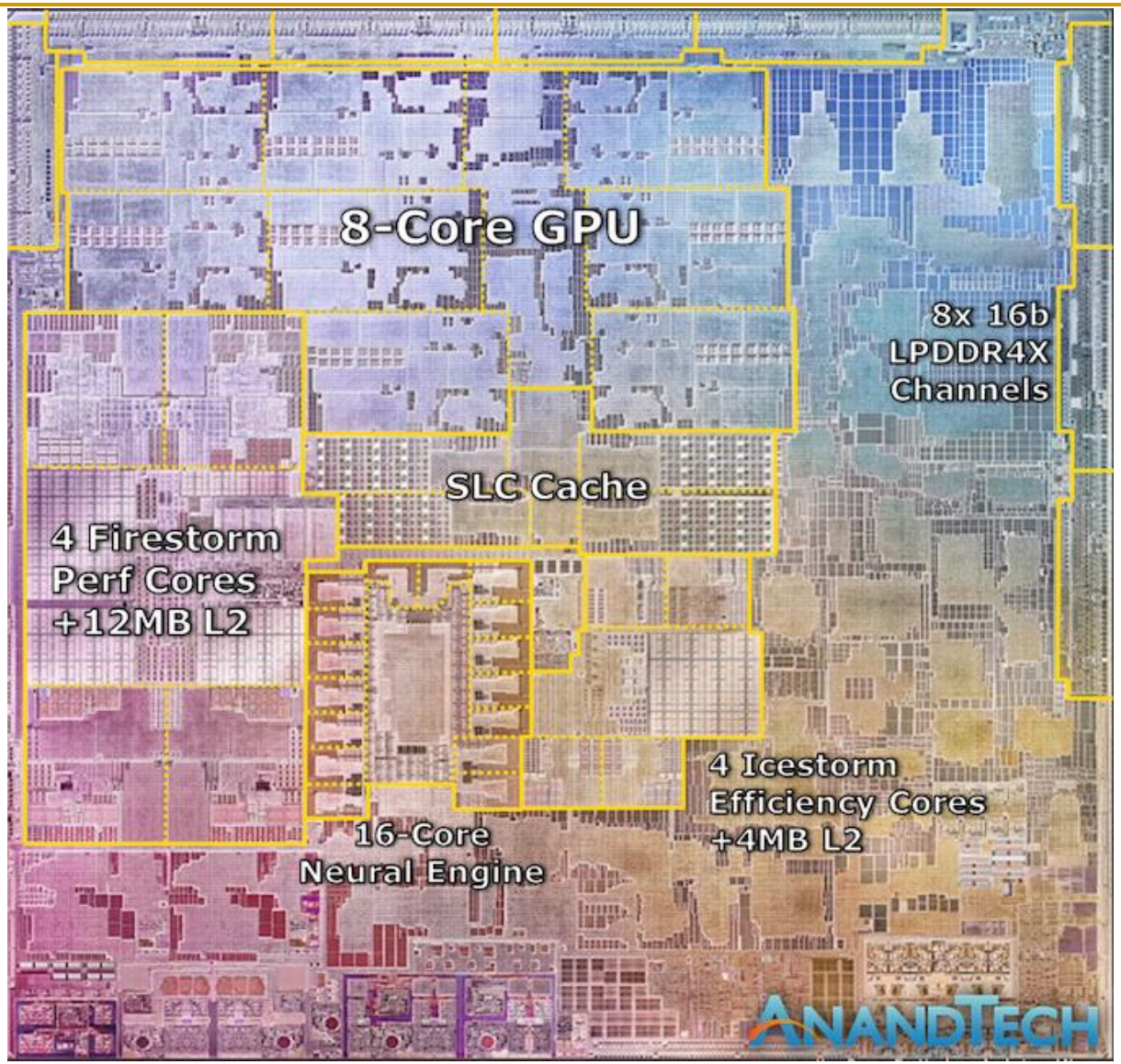
16 billion transistors

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



Source: https://www.apple.com/mac/m1/

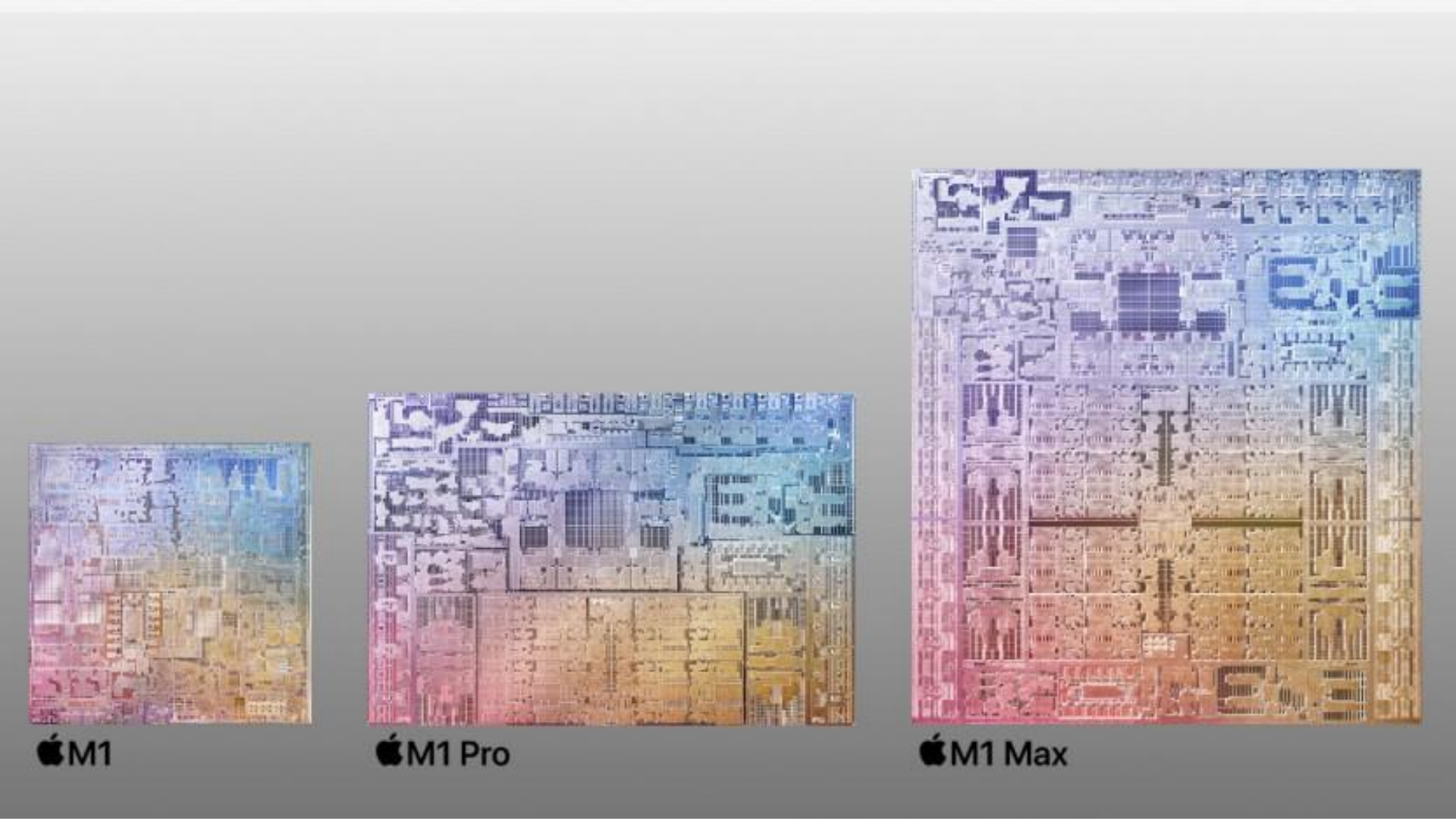
Modern General-Purpose Microprocessors



Apple M1, 2021

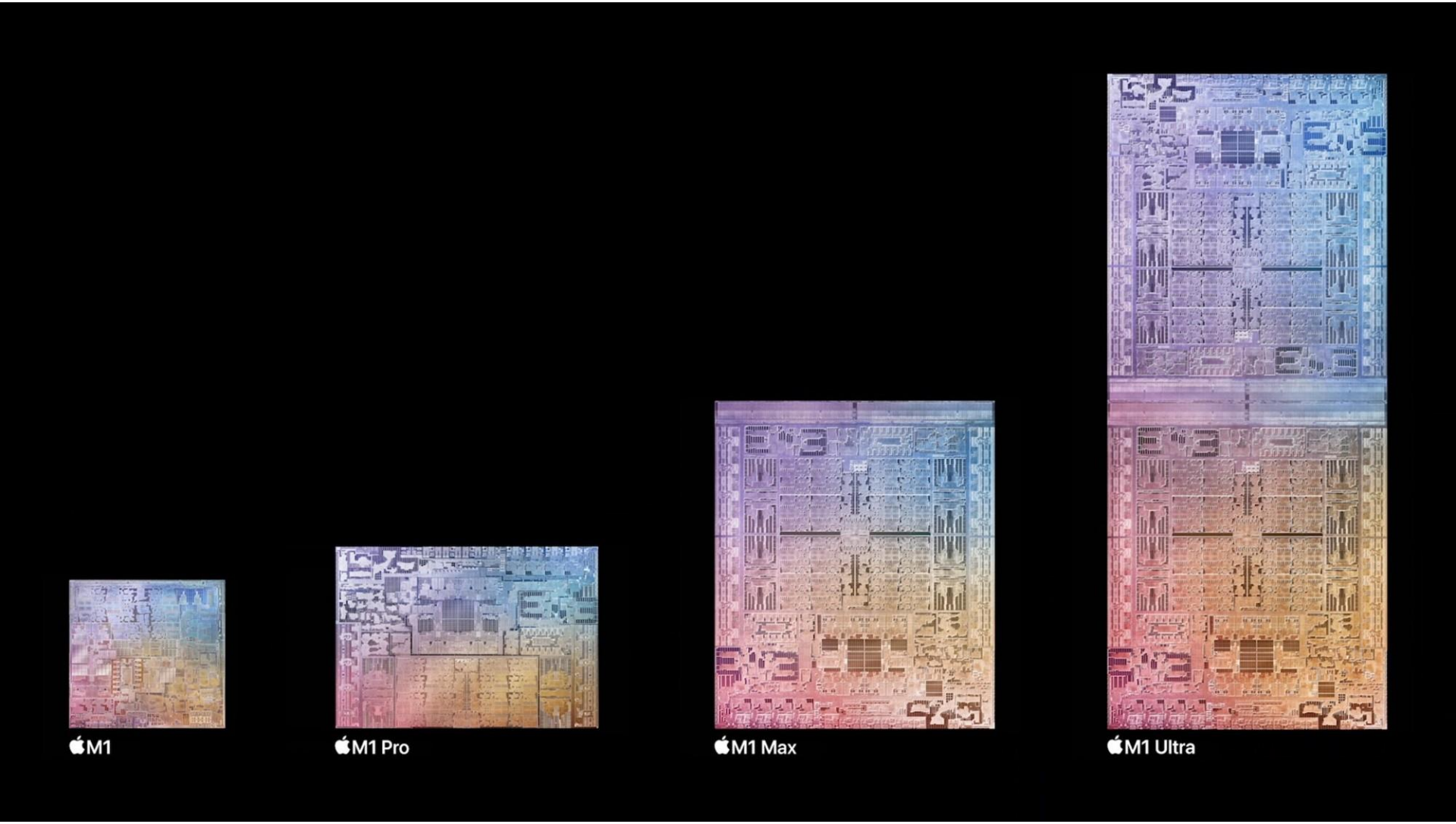
Source: https://www.anandtech.com/show/16252/mac-mini-apple-m1-tested

Modern General-Purpose Microprocessors



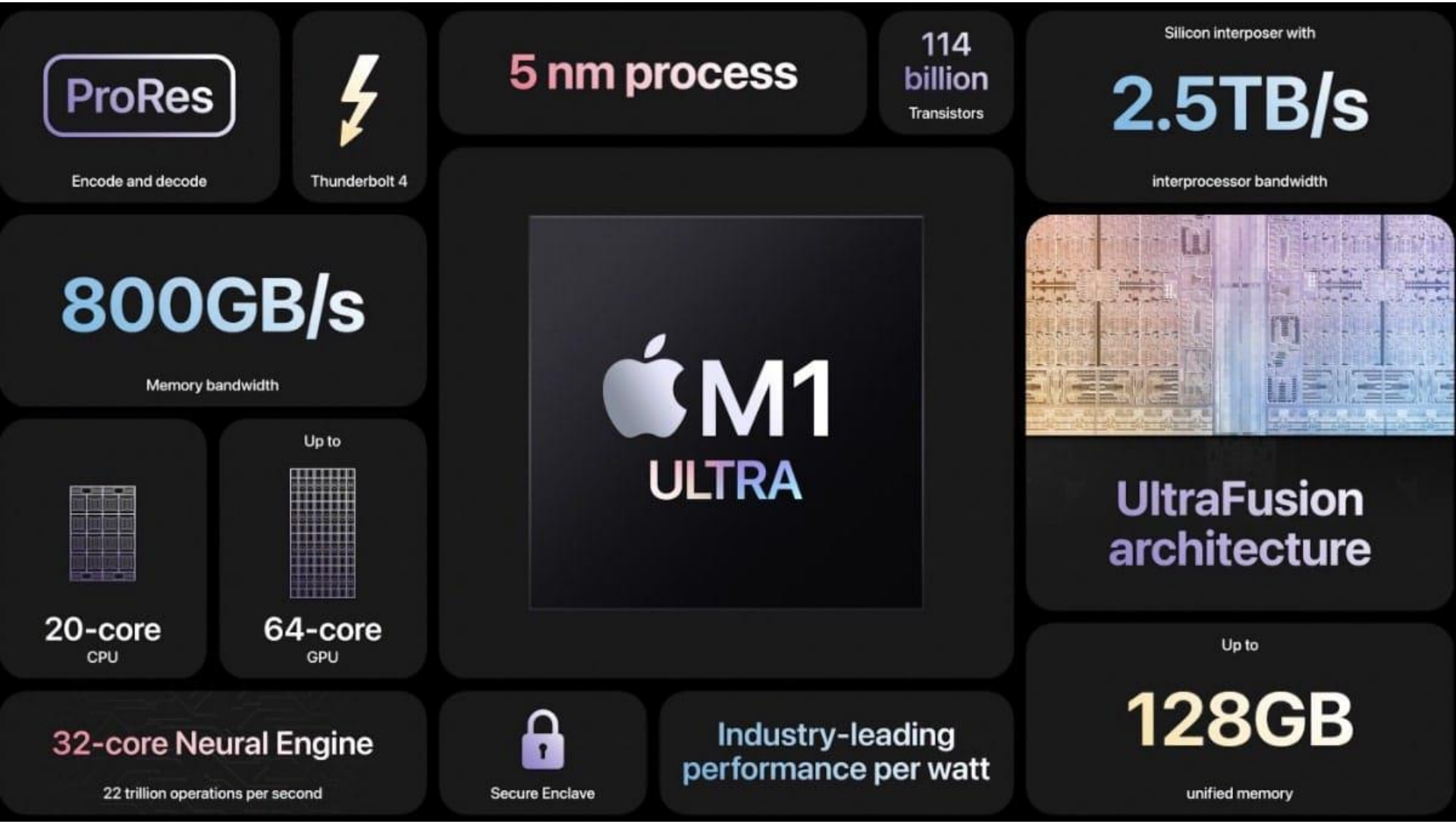
Source: https://www.golem.de/news/m1-pro-max-dieses-apple-silicon-ist-gigantisch-2110-160415.html

Apple M1 Ultra (2022)



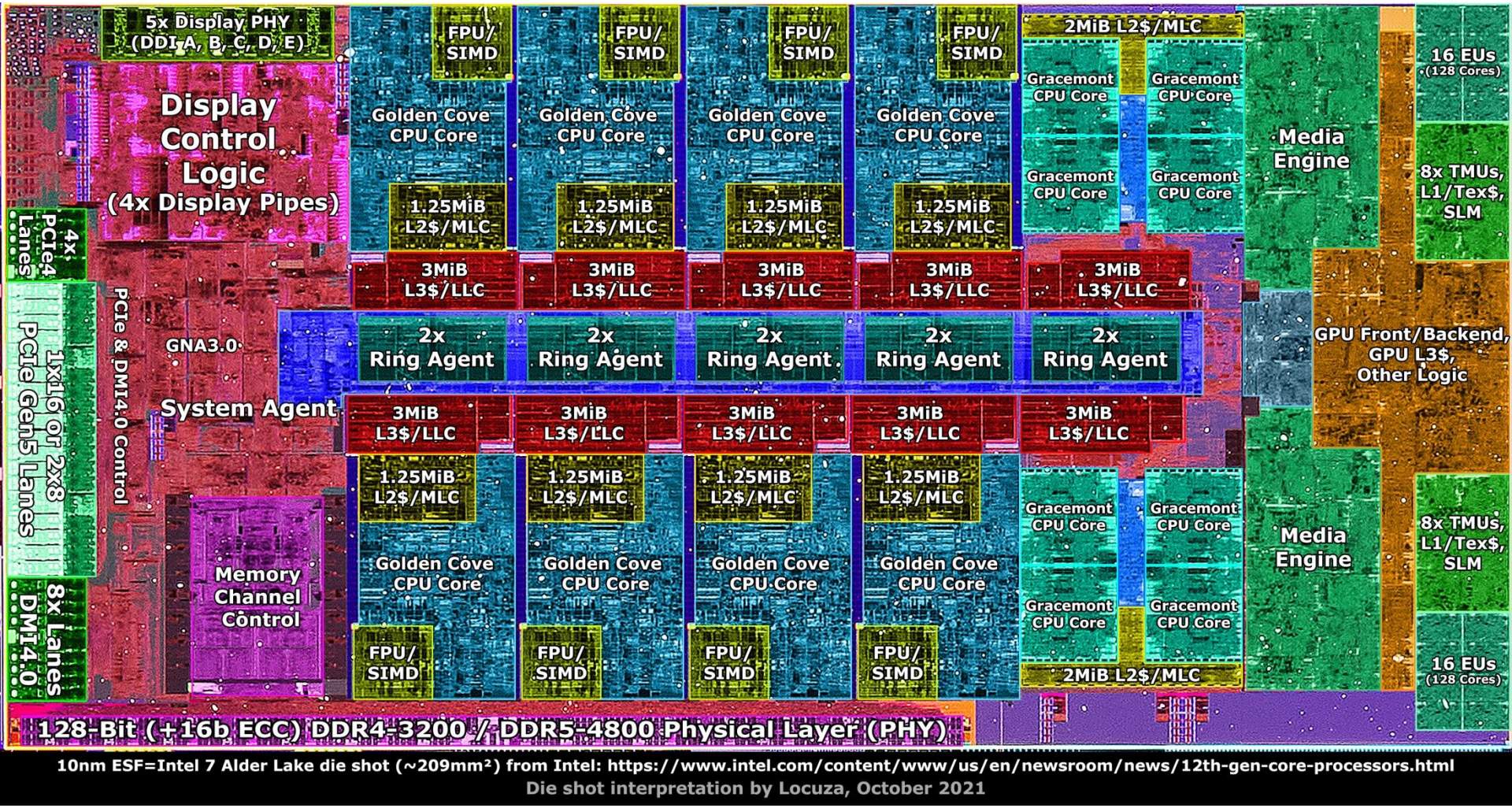
https://www.theverge.com/2022/3/9/22968611/apple-m1-ultra-gpu-nvidia-rtx-3090-comparison

Apple M1 Ultra (2022)



https://stadt-bremerhaven.de/apple-neuer-m1-ultra-chip-ist-offiziell/

Modern General-Purpose Microprocessors

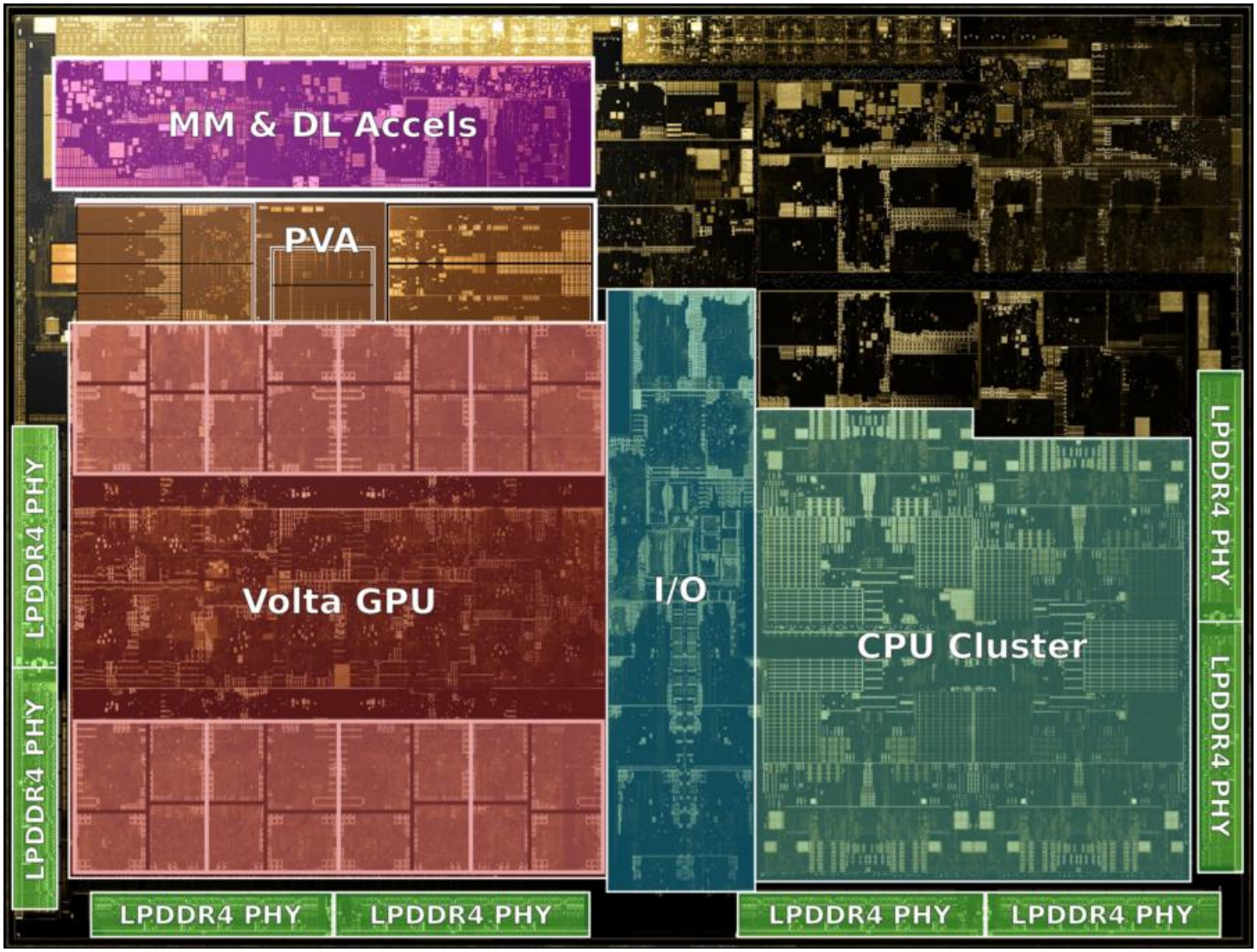


10nm ESF=Intel 7 Alder Lake die shot (~209mm²) from Intel: https://www.intel.com/content/www/us/en/newsroom/news/12th-gen-core-processors.html
Die shot interpretation by Locuza, October 2021

Intel Alder Lake, 2021

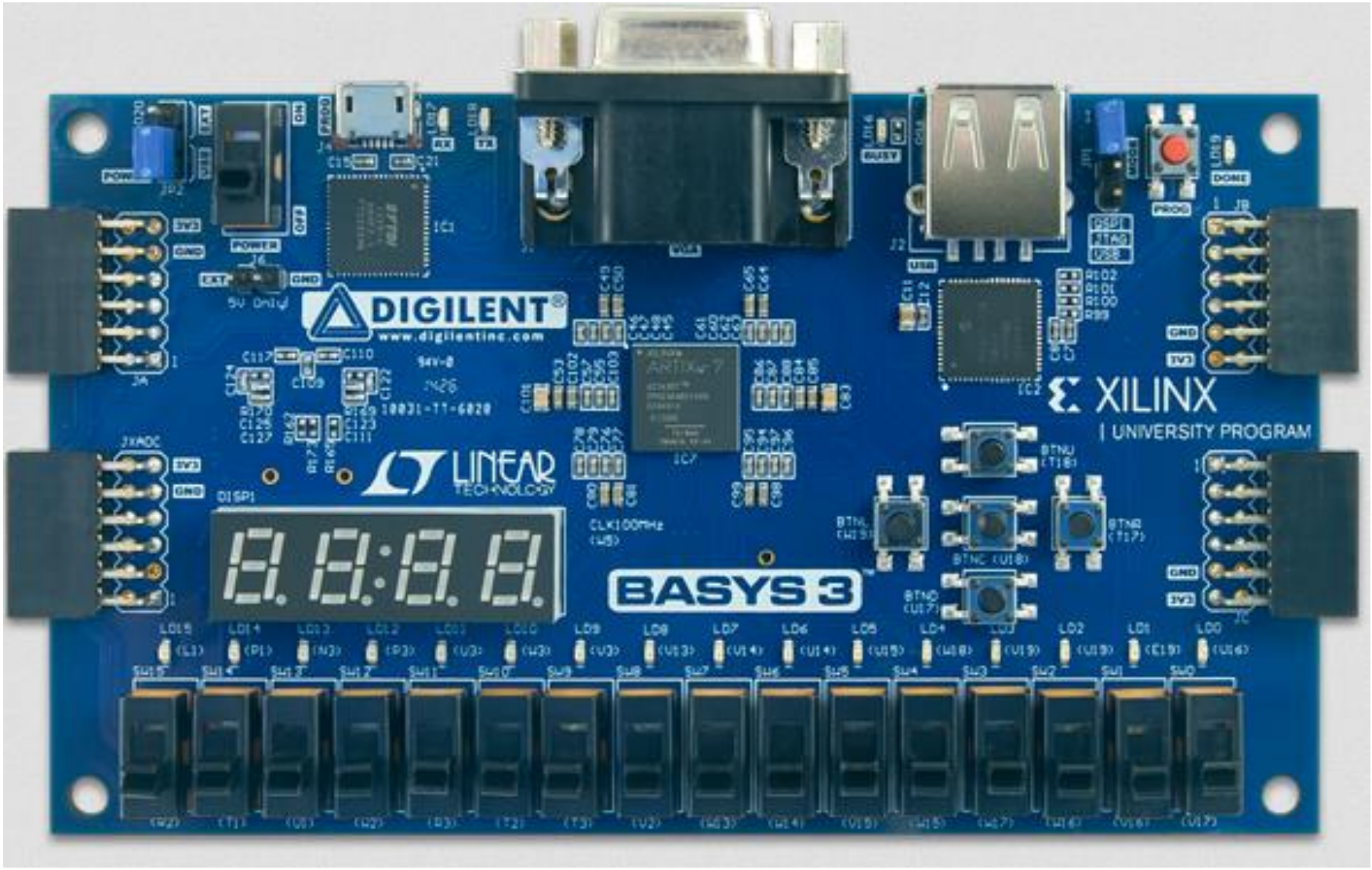
Source: https://twitter.com/Locuza_/status/1454152714930331652

Modern GPUs

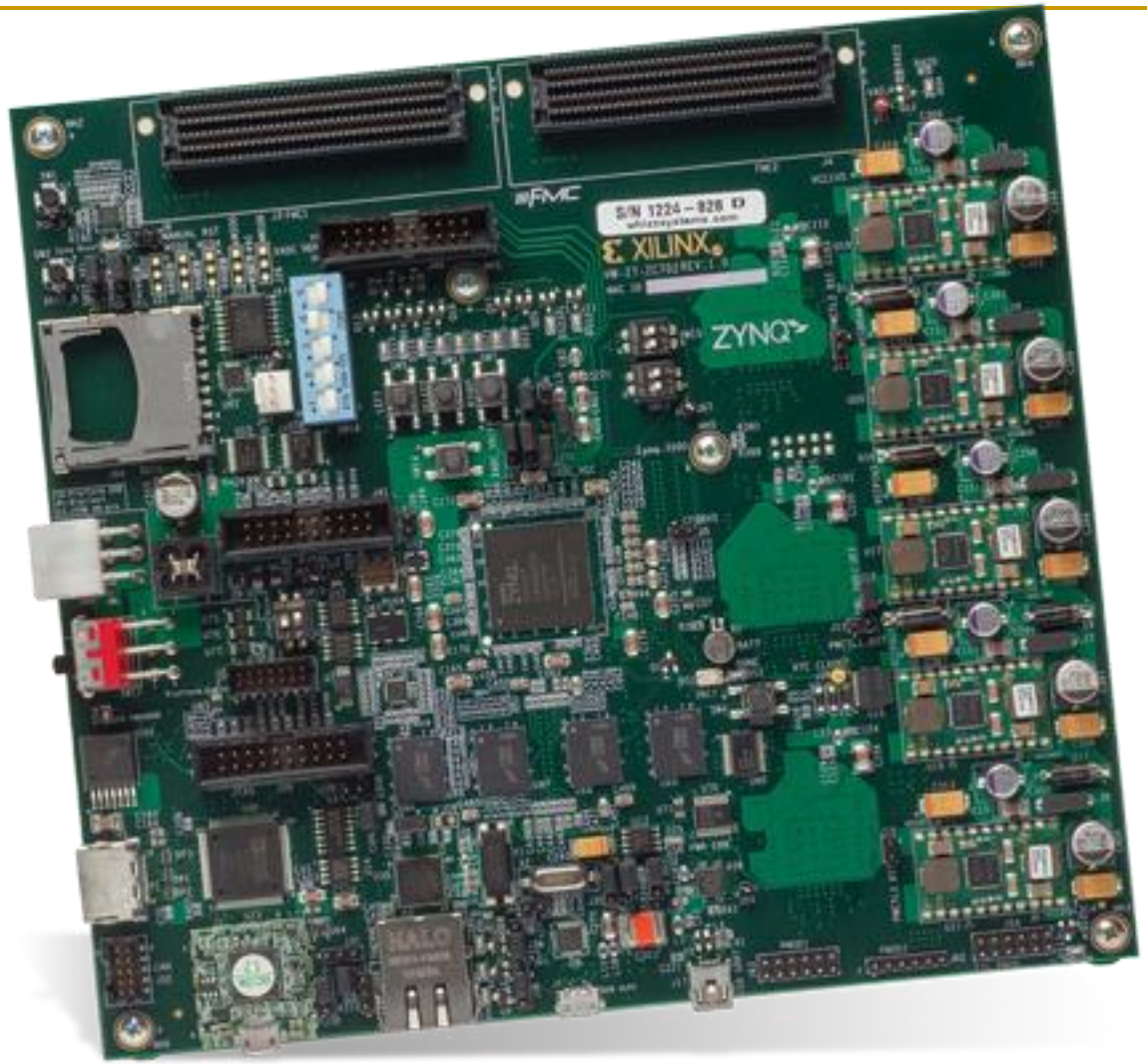


Source: https://en.wikichip.org/wiki/nvidia/tegra/xavier

FPGAs

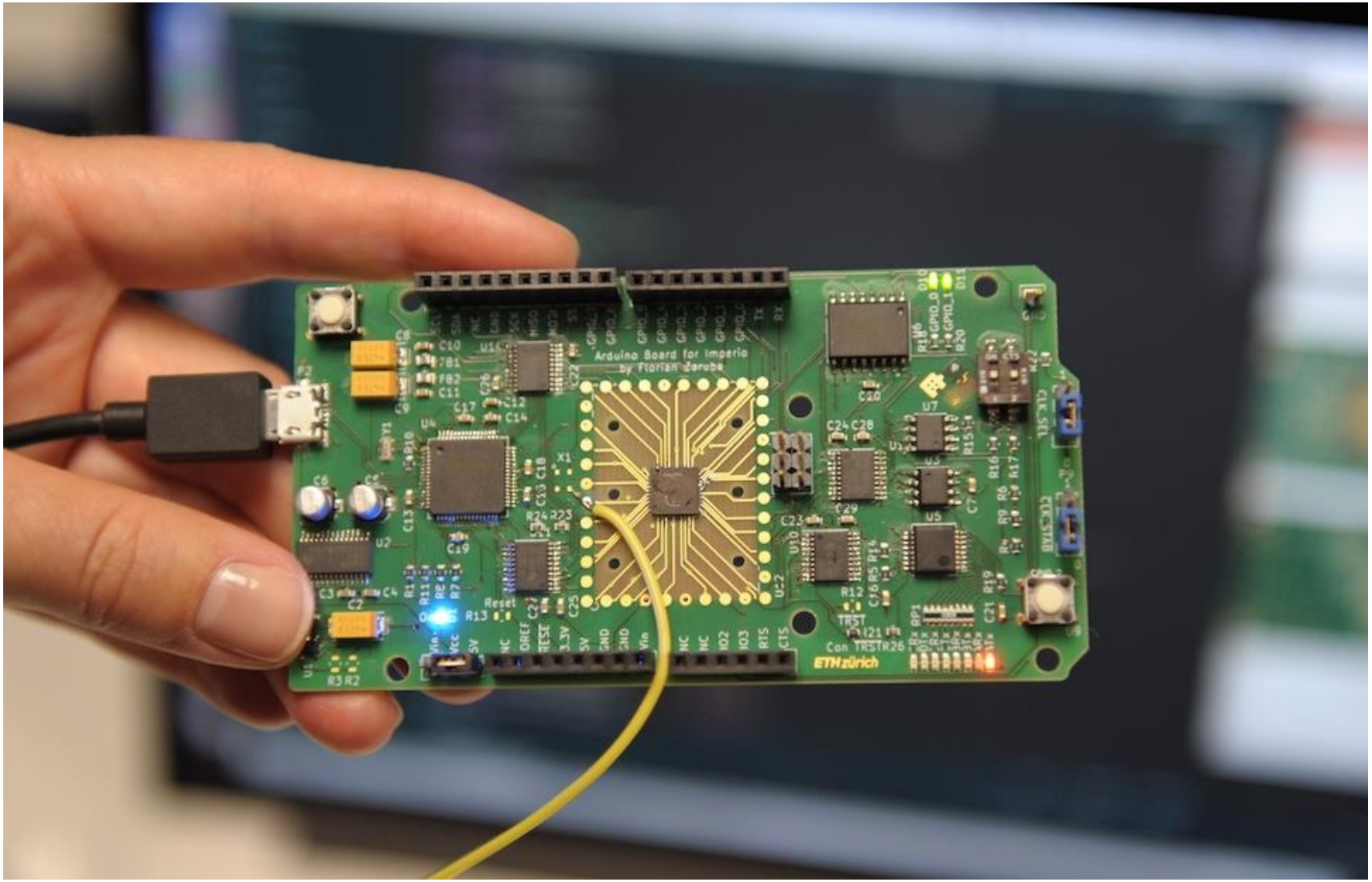

DIGILENT
www.digilentinc.com
XILINX
UNIVERSITY PROGRAM
LINEAR TECHNOLOGY
BASYS 3

Modern FPGAs



Source: https://www.mouser.ch/new/xilinx/xilinx-zynq-7000-zc702-eval-kit/

Special-Purpose ASICs (App-Specific Integrated Circuits)



Modern Special-Purpose ASICs

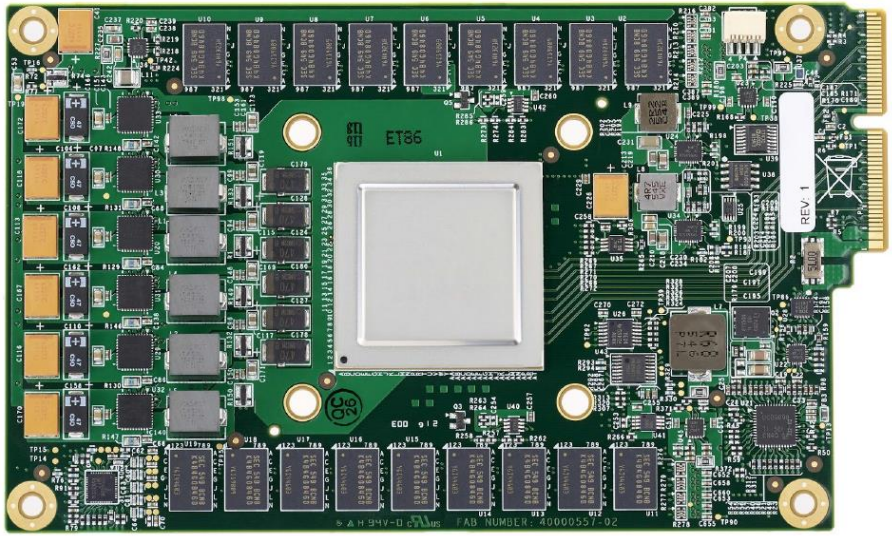


Figure 3. TPU Printed Circuit Board. It can be inserted in the slot for an SATA disk in a server, but the card uses PCIe Gen3 x16.

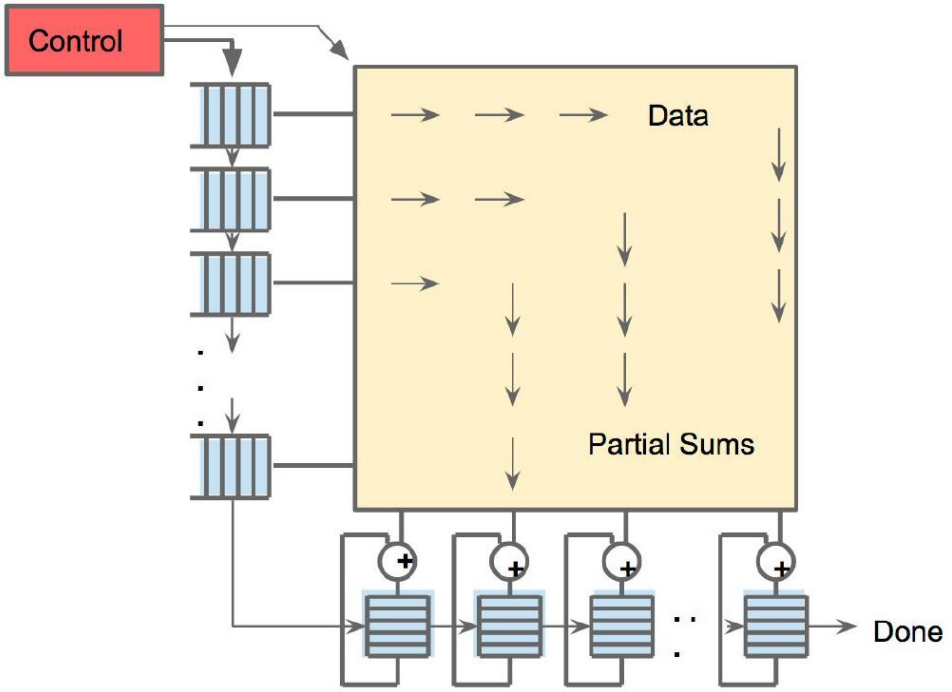
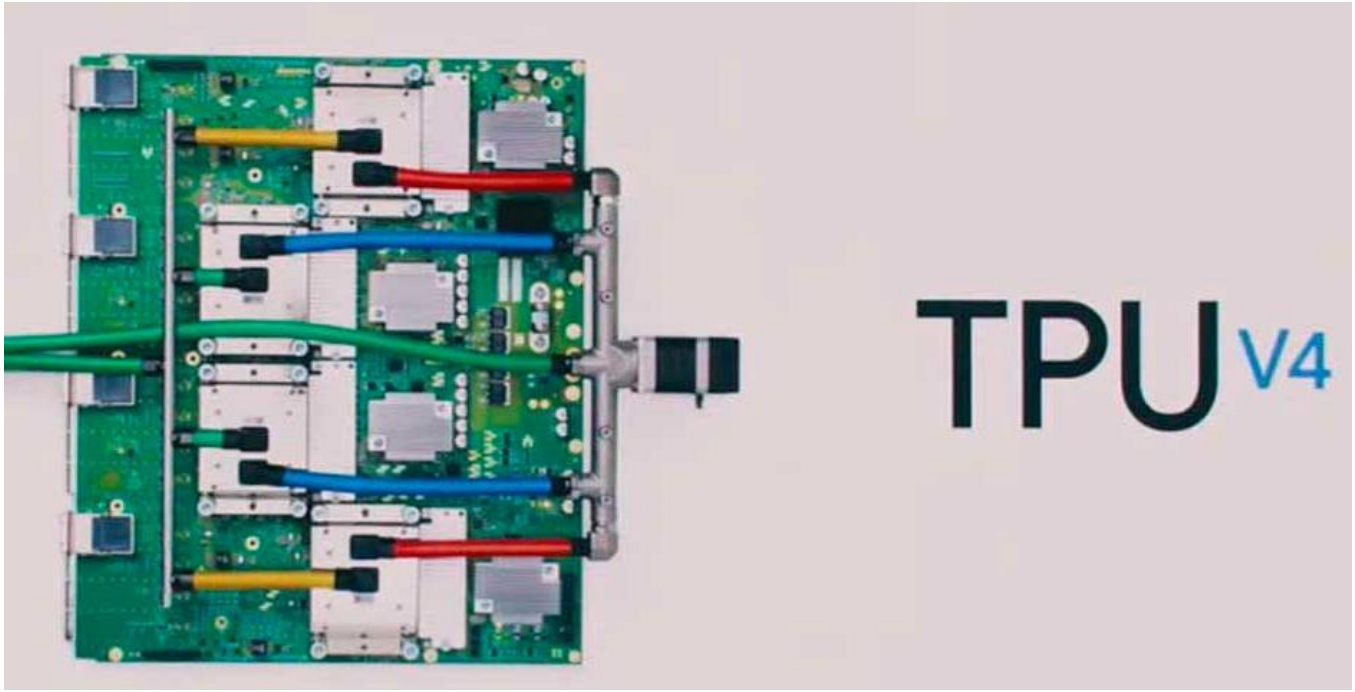


Figure 4. Systolic data flow of the Matrix Multiply Unit. Software has the illusion that each 256B input is read at once, and they instantly update one location of each of 256 accumulator RAMs.

Jouppi et al., "In-Datacenter Performance Analysis of a Tensor Processing Unit", ISCA 2017.

Modern Special-Purpose ASICs



New ML applications (vs. TPU3):

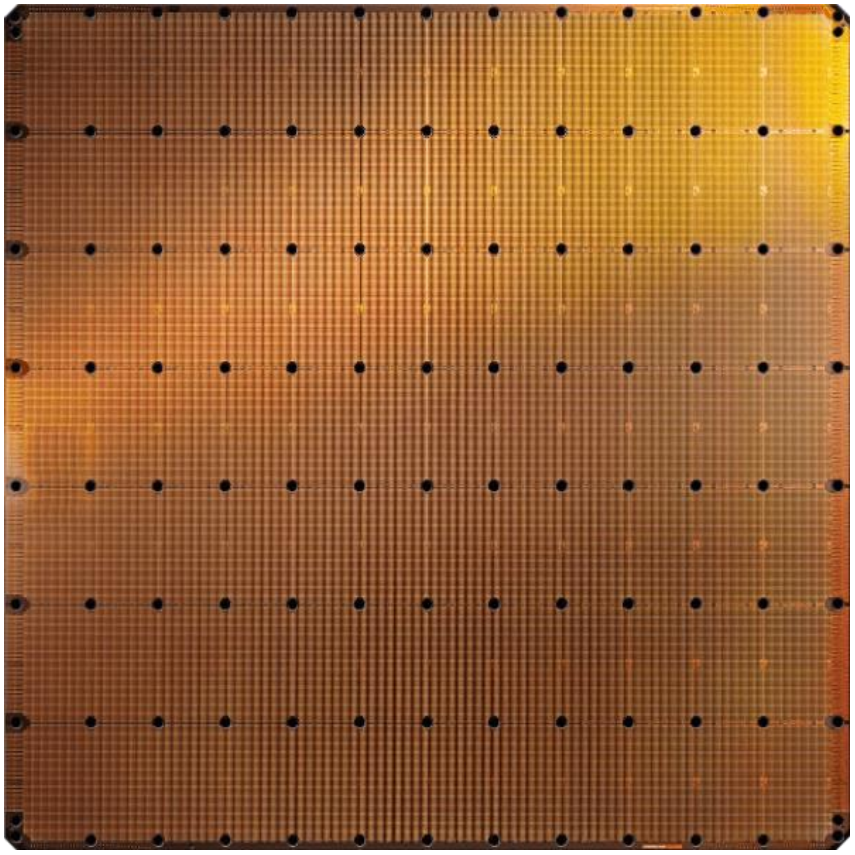
- Computer vision
- Natural Language Processing (NLP)
- Recommender system
- Reinforcement learning that plays Go

250 TFLOPS per chip in 2021
vs 90 TFLOPS in TPU3

1 ExaFLOPS per board

https://spectrum.ieee.org/tech-talk/computing/hardware/heres-how-googles-tpu-v4-ai-chip-stacked-up-in-training-tests

Modern Special-Purpose ASICs



- The largest ML accelerator chip (2021)
- 850,000 cores

Cerebras WSE-2
2.6 Trillion transistors
46,225 mm^2

Largest GPU
54.2 Billion transistors
826 mm^2

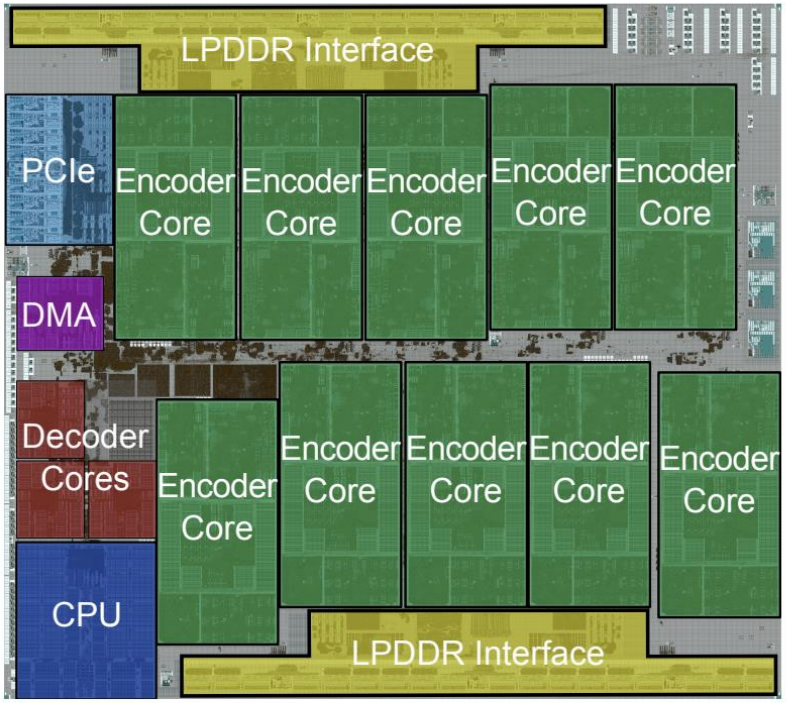
NVIDIA Ampere GA100

https://www.anandtech.com/show/14758/hot-chips-31-live-blogs-cerebras-wafer-scale-deep-learning

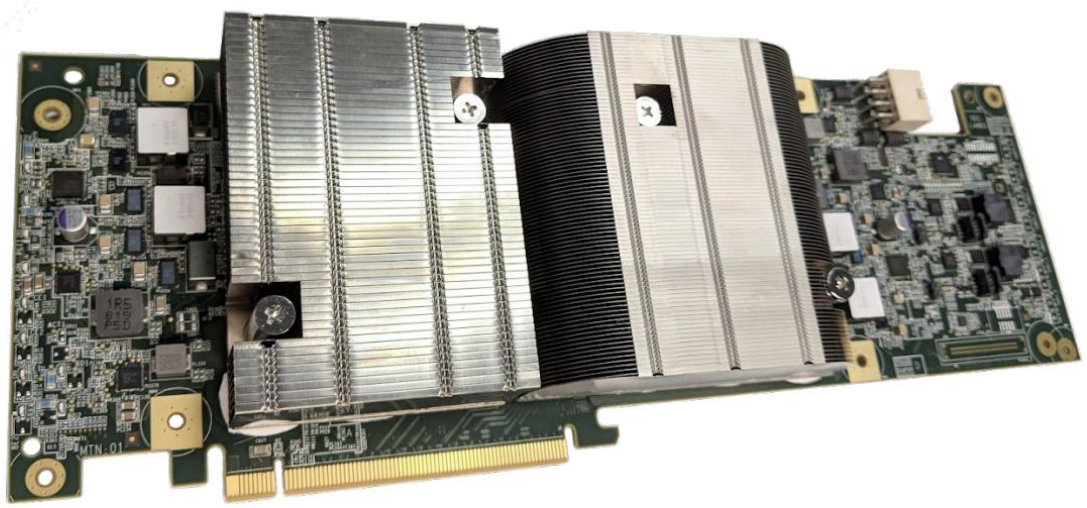
https://www.cerebras.net/cerebras-wafer-scale-engine-why-we-need-big-chips-for-deep-learning/

Modern Special-Purpose ASICs

Warehouse-Scale Video Acceleration: Co-design and Deployment in the Wild



(a) Chip floorplan



(b) Two chips on a PCBA

Figure 5: Pictures of the VCU

Source: https://dl.acm.org/doi/pdf/10.1145/3445814.3446723

They All Look the Same

	Microprocessors	FPGAs	ASICs
			
In short:	Common building block of computers	Reconfigurable hardware, flexible	You customize everything

They All Look the Same

	Microprocessors	FPGAs	ASICs
			
In short:	Common building block of computers	Reconfigurable hardware, flexible	You customize everything
Program Development Time	minutes	days	months

They All Look the Same

	Microprocessors	FPGAs	ASICs
			
In short:	Common building block of computers	Reconfigurable hardware, flexible	You customize everything
Program Development Time	minutes	days	months
Performance	o	+	++

They All Look the Same

	Microprocessors	FPGAs	ASICs
			
In short:	Common building block of computers	Reconfigurable hardware, flexible	You customize everything
Program Development Time	minutes	days	months
Performance	o	+	++
Good for	Ubiquitous Simple to use	Prototyping Small volume	Mass production, Max performance

They All Look (Kind of) the Same

	Microprocessors	FPGAs	ASICs
			
In short:	Common building block of computers	Reconfigurable hardware, flexible	You customize everything
Program Development Time	minutes	days	months
Performance	o	+	++
Good for	Ubiquitous Simple to use	Prototyping Small volume	Mass production, Max performance
Programming	Executable file	Bit file	Design masks
Languages	C/C++/Java/…	Verilog/VHDL	Verilog/VHDL
Main Companies	Intel, ARM, AMD, Apple, NVIDIA	Xilinx, Altera	TSMC, Globalfoundries

Labs: Build A Microprocessor on FPGA

Want to learn how these work

By programming these

	Microprocessors	FPGAs	
			
In short	Common building block of computers	Reconfigurable hardware, flexible	nize ng
Program Development Time	minutes	days	months
Performance	o	+	++
Good for	Ubiquitous Simple to use	Prototyping	Mass production.
Programming	Executable file	Using this language	
Languages	C/C++/Java/…	Verilog/VHDL	Verilog/VHDL
Main Companies	Intel, ARM, AMD, Apple, NVIDIA	Xilinx, Altera	TSMC, Globalfoundries

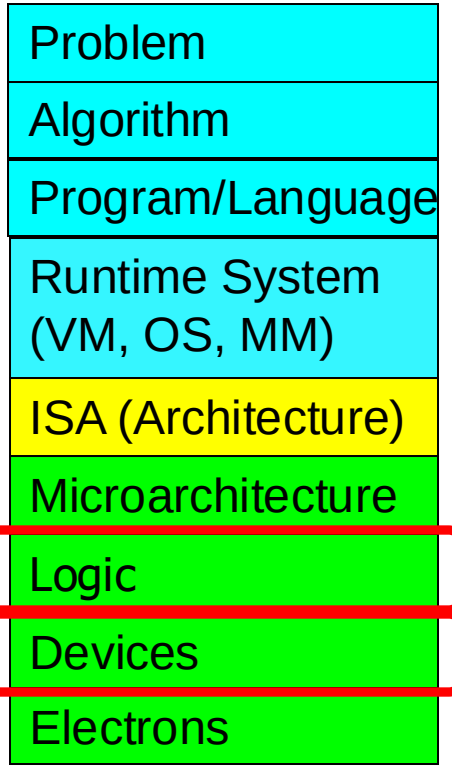
All Computers are Built Upon the Same Building Blocks

Building Blocks of Modern Computers

Transistors

Transistors

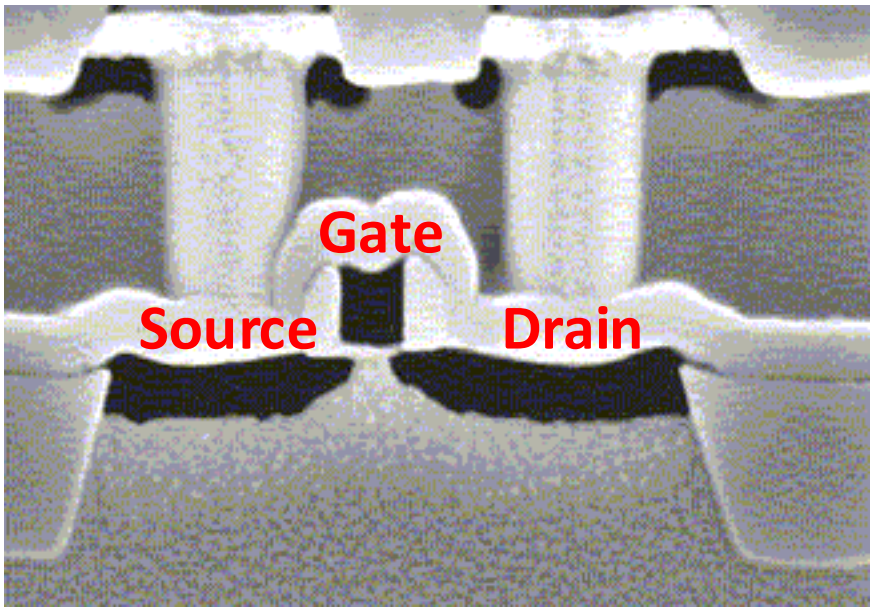
- **Computers are built from very large numbers of very small (and relatively simple) structures: transistors**
  - Intel 4004, in 1971, had 2300 MOS transistors
  - Intel's Pentium IV microprocessor, 2000, was made up of more than 42 **M**illion MOS transistors
  - Apple's M2 Max, offered for sale in 2022, is made up of more than 67 **B**illion MOS transistors



- **This lecture**
  - How the MOS transistor works (as a logic element)
  - How these transistors are connected to form logic gates
  - How logic gates are interconnected to form larger units that are needed to construct a computer

MOS Transistor

- By combining
  - Conductors (**M**etal)
  - Insulators (**O**xide)
  - **S**emiconductors

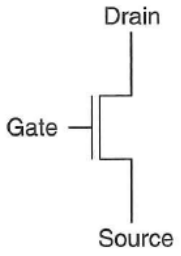


- We get a Transistor (MOS)
- Why is this useful?
  - We can combine many of these to realize simple logic gates
- The electrical properties of metal-oxide semiconductors are well beyond the scope of what we want to understand in this course
  - They are below our lowest level of abstraction

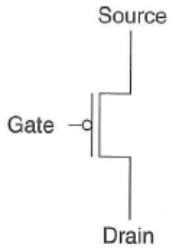
Sections 1.6 and 1.7 in Harris and Harris provide some lower-level explanations

Different Types of MOS Transistors

- There are two types of MOS transistors: n-type and p-type



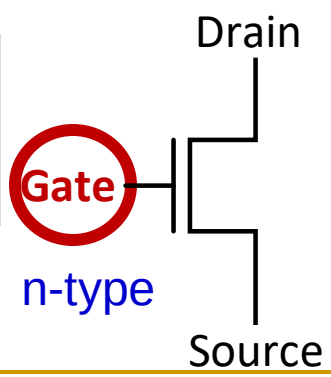
n-type



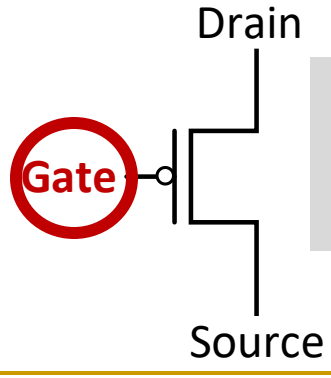
p-type

- They both operate "logically," very similar to the way wall switches work

The circuit is closed when the gate is supplied with 3V



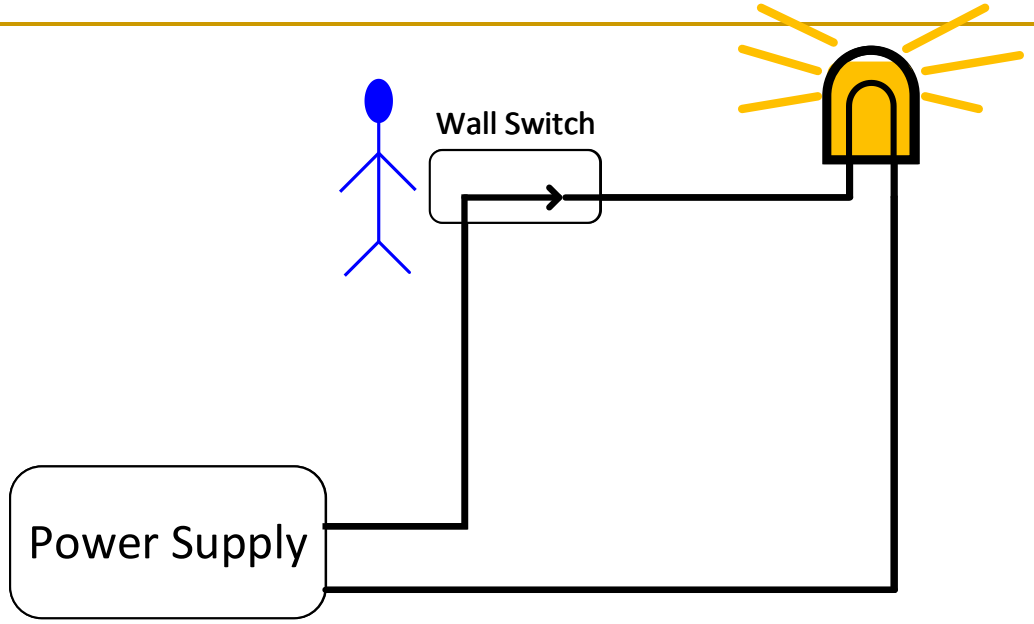
n-type



The circuit is closed when the gate is supplied with 0V

p-type

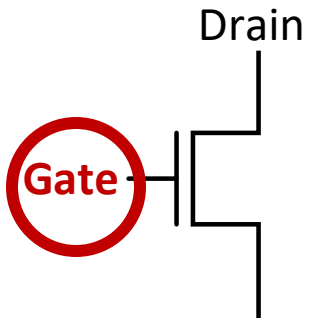
How Does a Wall Switch Work?



- In order for the lamp to glow, electrons must flow
- In order for electrons to flow, there must be a closed circuit from the power supply to the lamp and back to the power supply
- The lamp can be turned on and off by simply manipulating the wall switch to make or break the closed circuit

Transistor Works As a Switch

- Instead of the wall switch, we could use an n-type or a p-type MOS transistor to make or break the closed circuit



Schematic of an n-type MOS transistor

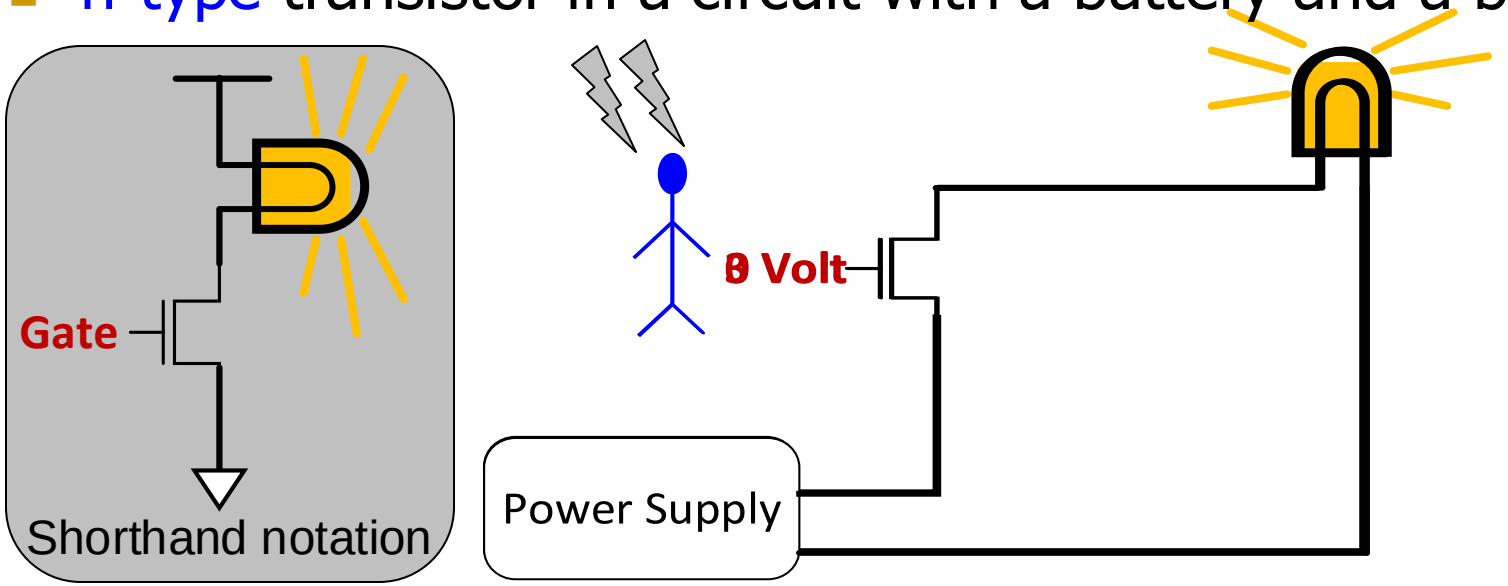
If the gate of an n-type transistor is supplied with a **high** voltage, the connection from source to drain acts like a piece of wire (i.e., the circuit is closed)

Depending on the technology, high voltage can range from 0.3V to 3V

If the gate of the n-type transistor is supplied with **zero** voltage, the connection between the source and drain is broken (i.e., the circuit is open)

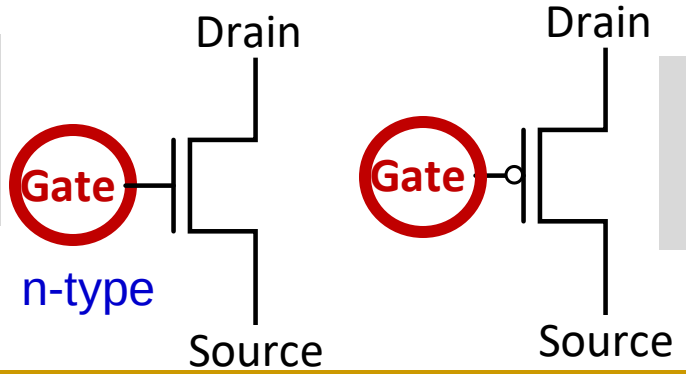
Abstraction: Transistor As a Switch

- n-type transistor in a circuit with a battery and a bulb



- p-type transistor works in exactly the opposite fashion from n-type transistor

The circuit is closed when the gate is supplied with 3V



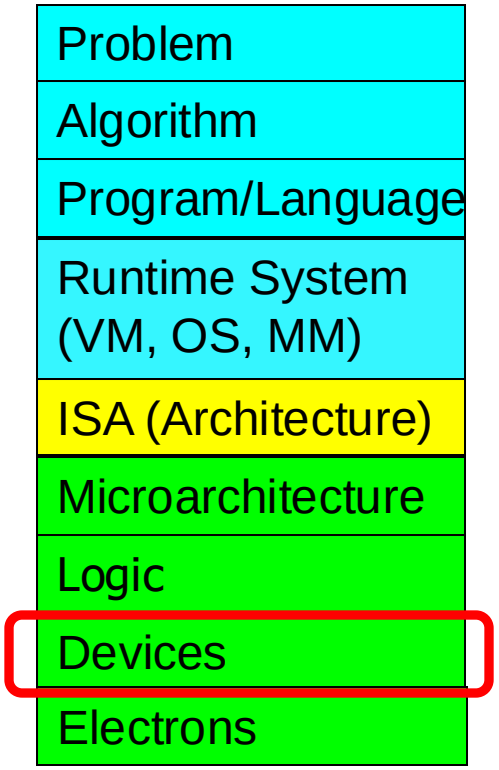
The circuit is closed when the gate is supplied with 0V

p-type

Logic Gates

One Level Higher in the Abstraction

- **Now, we know how a MOS transistor works**
- How do we build logic structures out of MOS transistors?
- We construct basic logical units out of individual MOS transistors
- These logical units are called logic gates
  - They implement simple Boolean functions



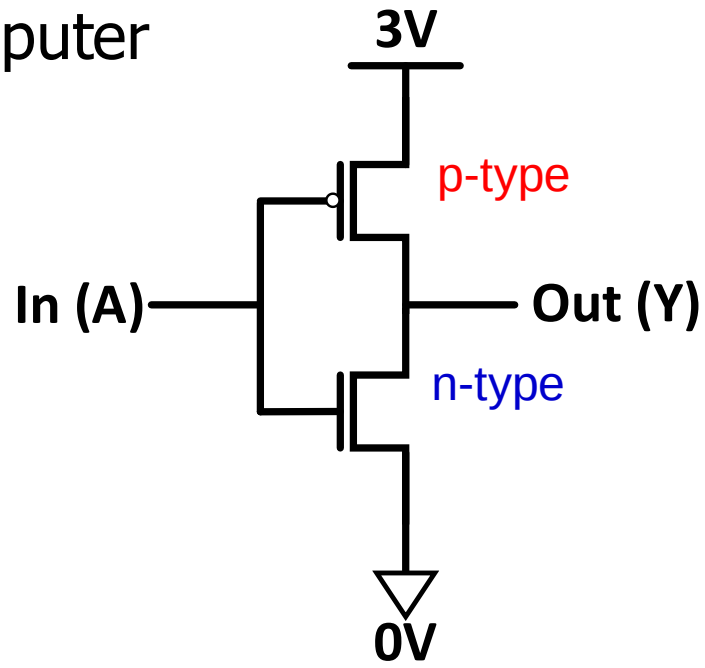
George Boole, "The Mathematical Analysis of Logic," 1847.

Making Logic Blocks Using CMOS Technology

- Modern computers use both n-type and p-type transistors, i.e. Complementary MOS (CMOS) technology

nMOS + pMOS = CMOS

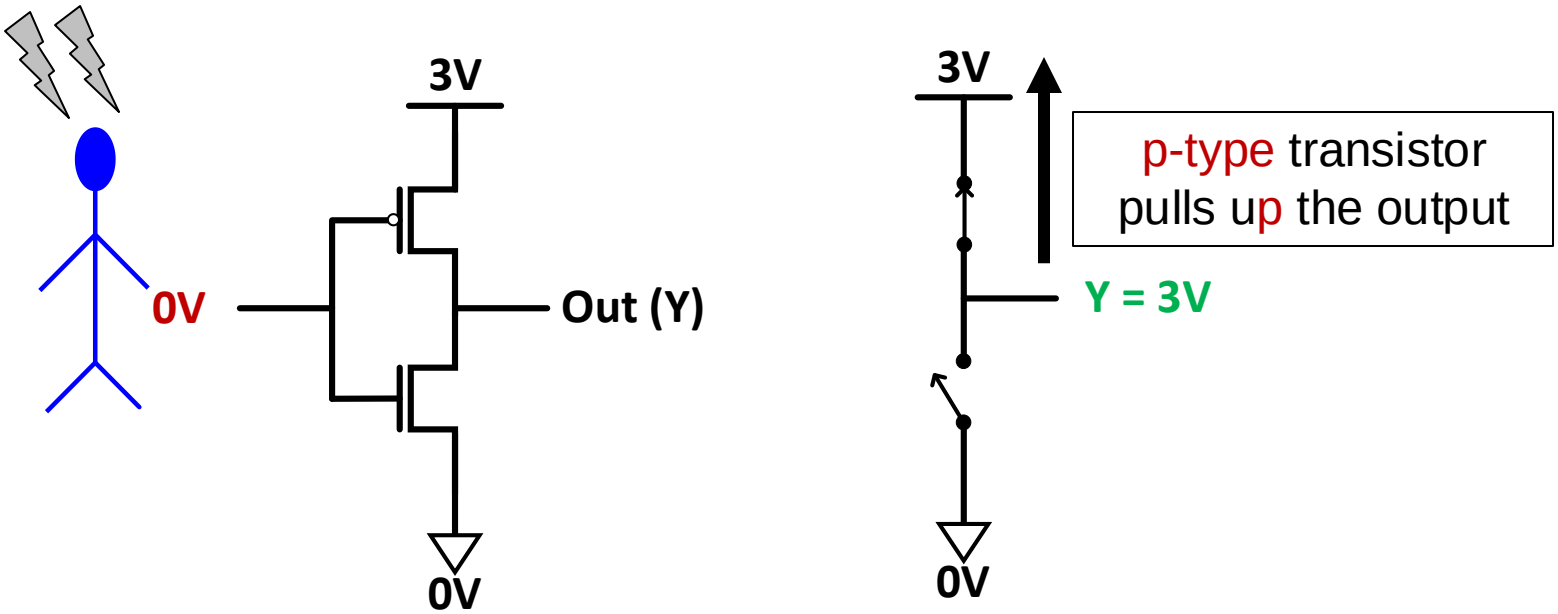
- The simplest logic structure that exists in a modern computer



What does this circuit do?

Functionality of Our CMOS Circuit

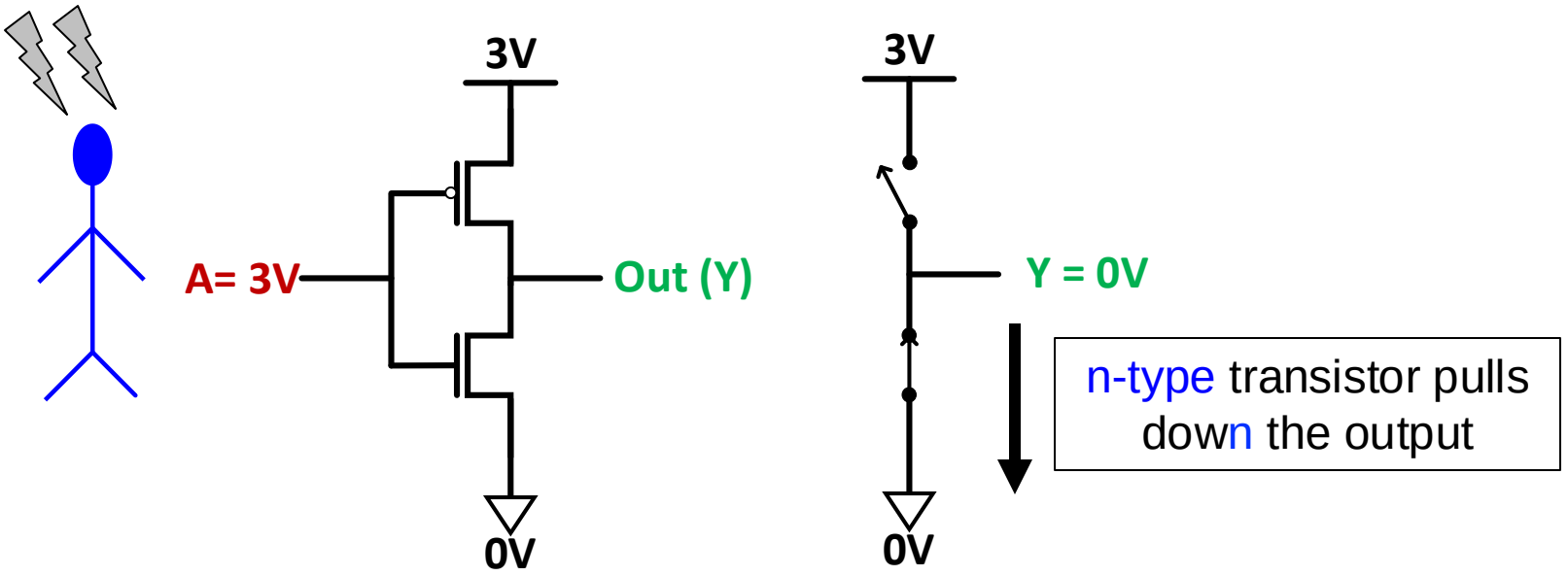
What happens when the input is connected to 0V?



p-type transistors are good at pulling u**p** the voltage

Functionality of Our CMOS Circuit

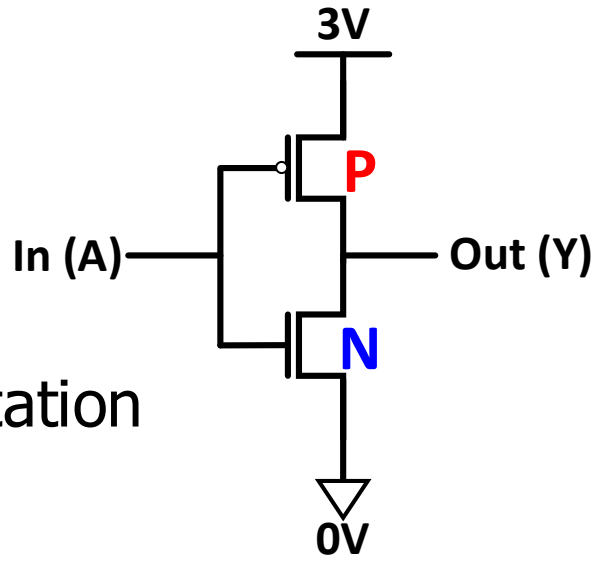
What happens when the input is connected to 3V?



n-type transistors are good at pulling down the voltage

CMOS NOT Gate (Inverter)

- This is actually the CMOS NOT Gate
- Why do we call it NOT?
  - If A = 0V then Y = 3V
  - If A = 3V then Y = 0V
- **Digital circuit:** one possible interpretation
  - Interpret 0V as logical (binary) 0 value
  - Interpret 3V as logical (binary) 1 value

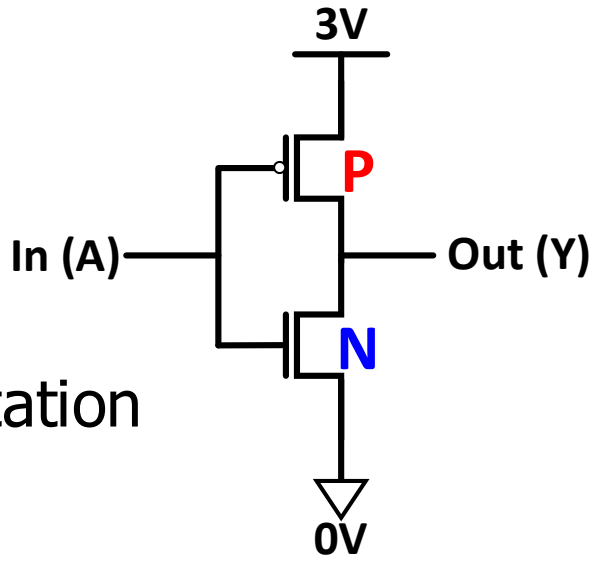


A	P	N	Y
0	**ON**	OFF	1
1	OFF	**ON**	0

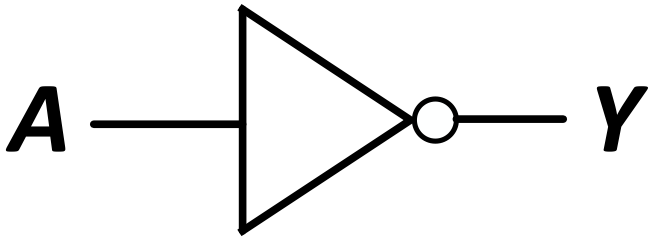
$$Y = \overline{A}$$

CMOS NOT Gate (Inverter)

- This is actually the CMOS NOT Gate
- Why do we call it NOT?
  - If A = 0V then Y = 3V
  - If A = 3V then Y = 0V
- **Digital circuit:** one possible interpretation
  - Interpret 0V as logical (binary) 0 value
  - Interpret 3V as logical (binary) 1 value



$$Y = \overline{A}$$



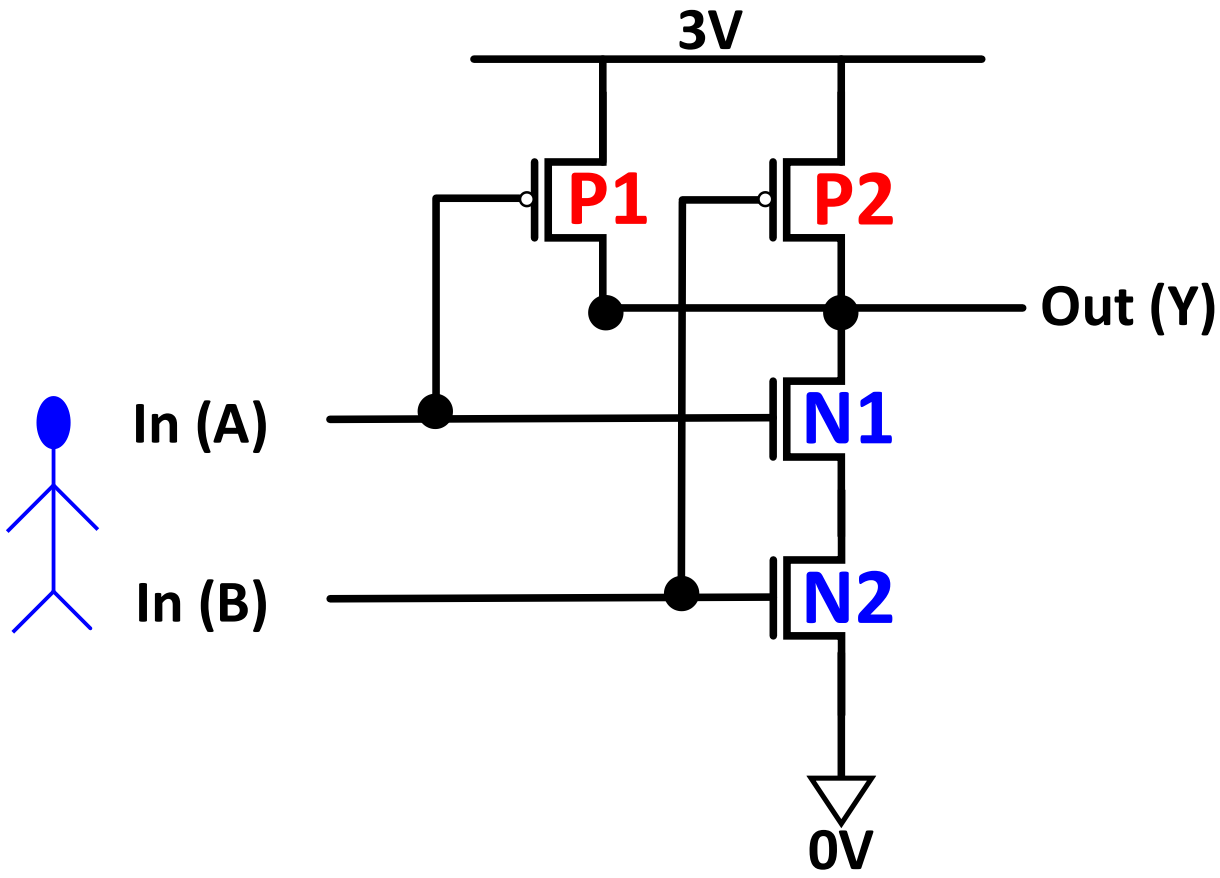
We call this a NOT gate
or an inverter
(bubble indicates inversion)

Truth table: shows the logical output of the circuit for each possible input

A	Y
0	1
1	0

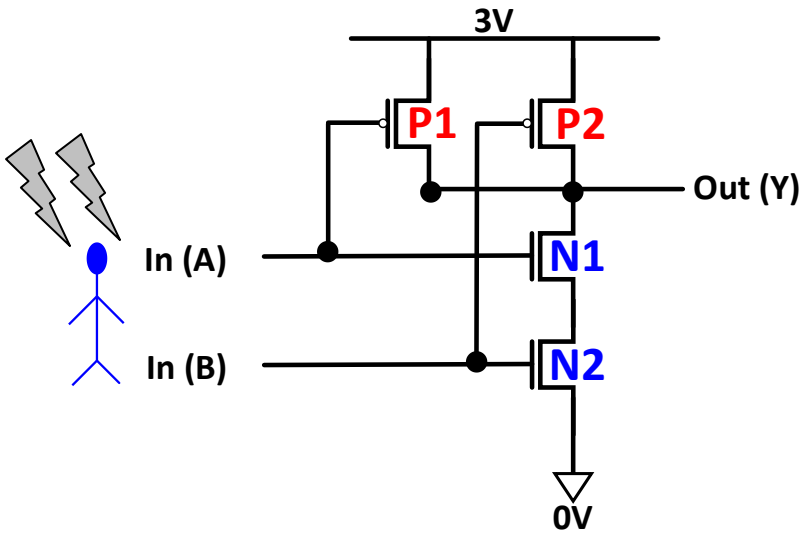
Another CMOS Gate: What Is This?

- Let's build more complex gates!



CMOS NAND Gate

- Let's build more complex gates!



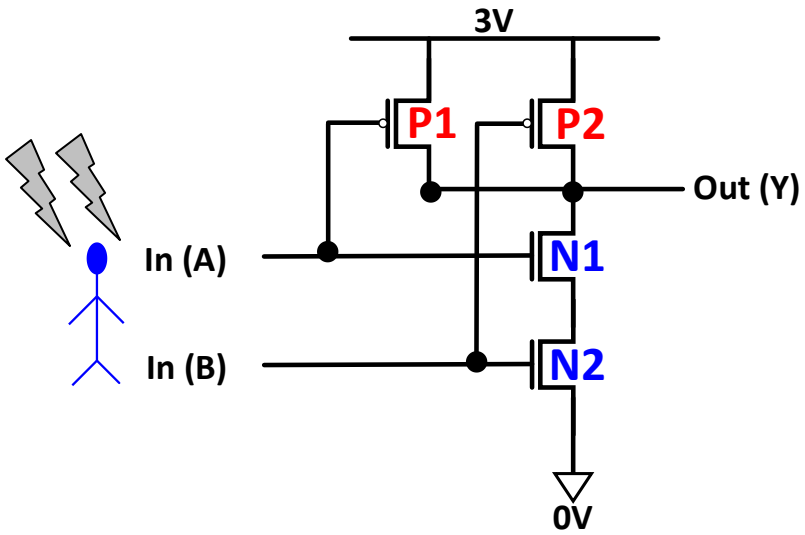
$$Y = \overline{A \cdot B} = \overline{AB}$$

A	B	P1	P2	N1	N2	Y
0	0	**ON**	**ON**	OFF	OFF	1
0	1	**ON**	OFF	OFF	ON	1
1	0	OFF	**ON**	ON	OFF	1
1	1	OFF	OFF	**ON**	**ON**	0

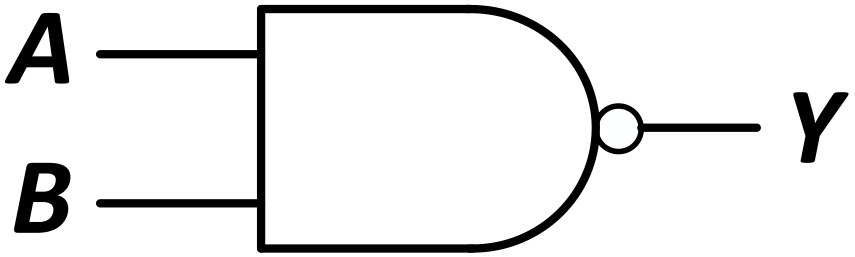
- P1 and P2 are in parallel; **only one must be ON to pull up the output to 3V**
- N1 and N2 are connected in series; **both must be ON to pull down the output to 0V**

CMOS NAND Gate

- Let's build more complex gates!



$$Y = \overline{A \cdot B} = \overline{AB}$$



A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

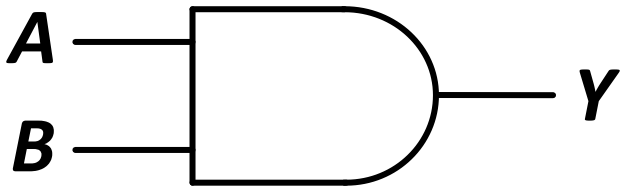
We call this a NAND gate
(bubble indicates inversion)

CMOS AND Gate

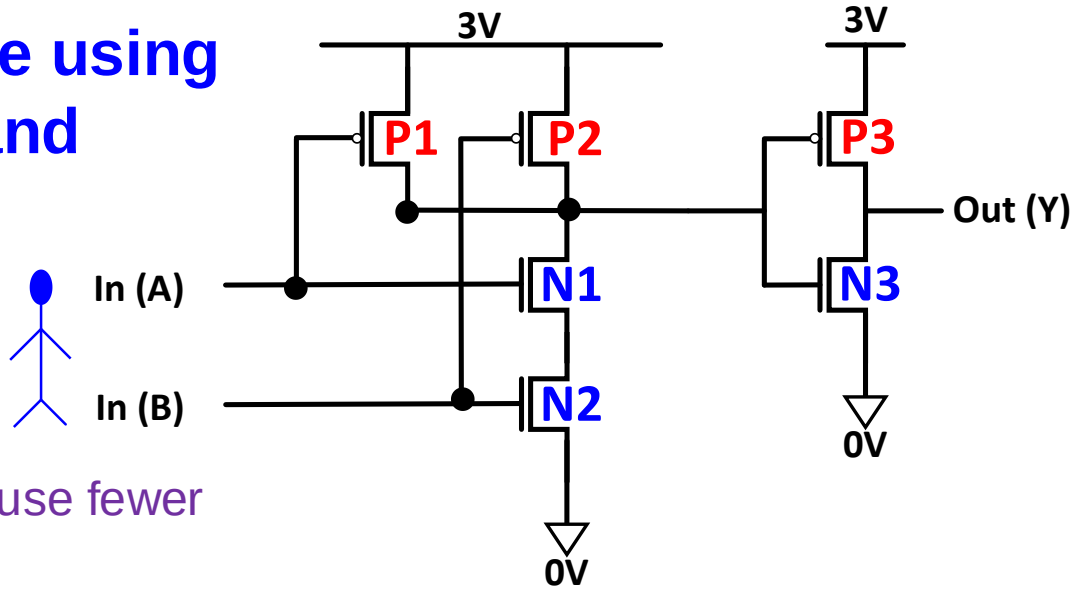
- How can we make an AND gate?

$$Y = A \cdot B = AB$$

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

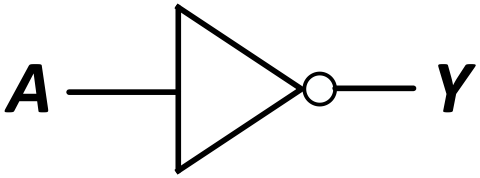


We make an AND gate using one NAND gate and one NOT gate

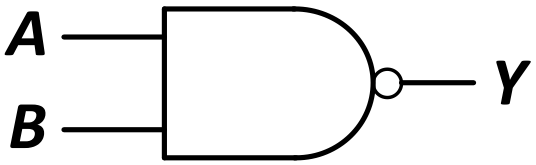


Food for thought: Can we not use fewer transistors for the AND gate?

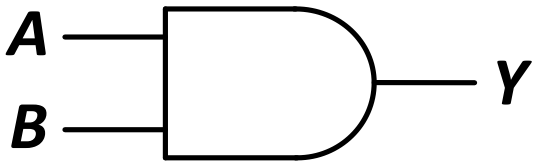
CMOS NOT, NAND, AND Gates



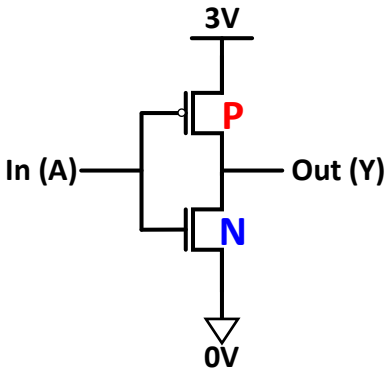
A	Y
0	1
1	0

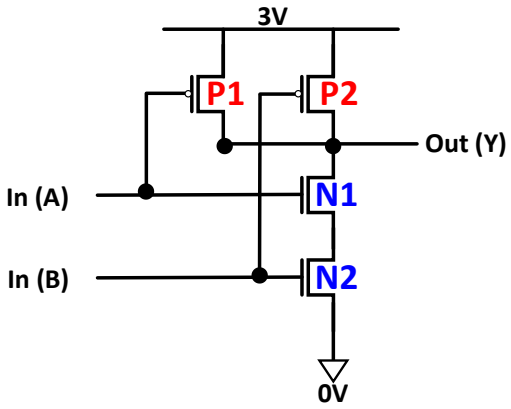


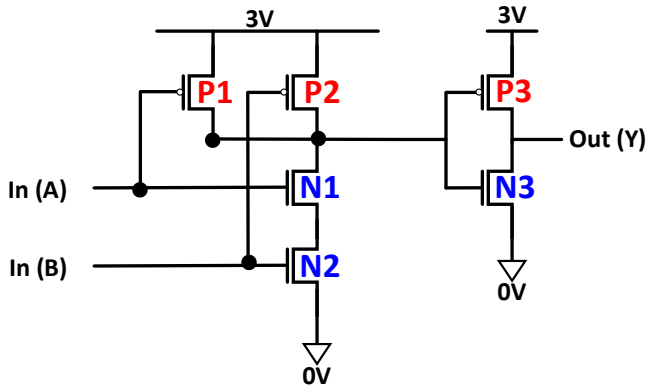
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0



A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

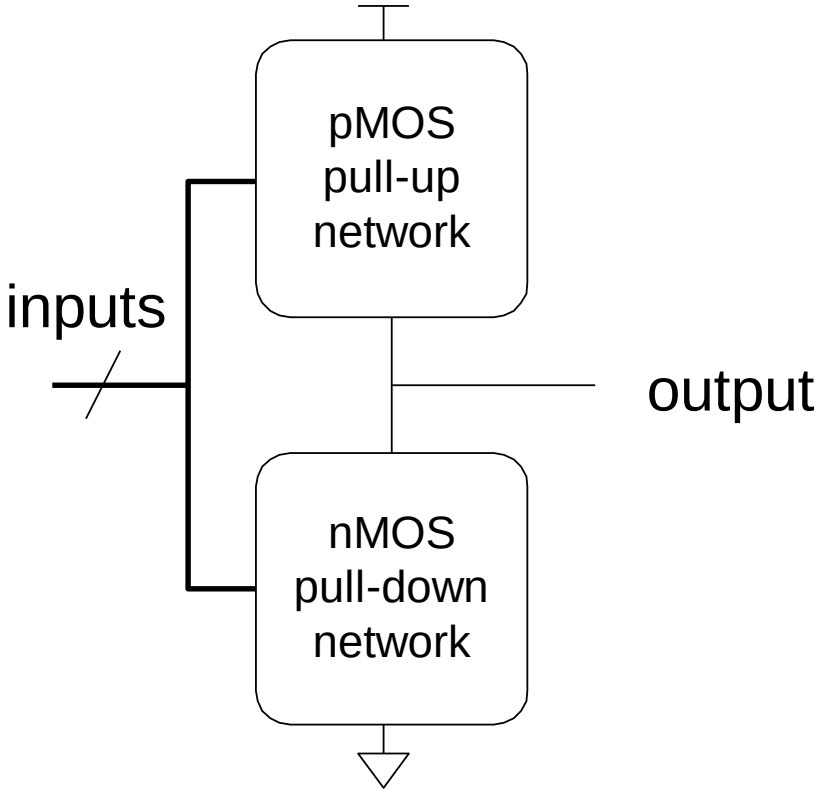






General CMOS Gate Structure

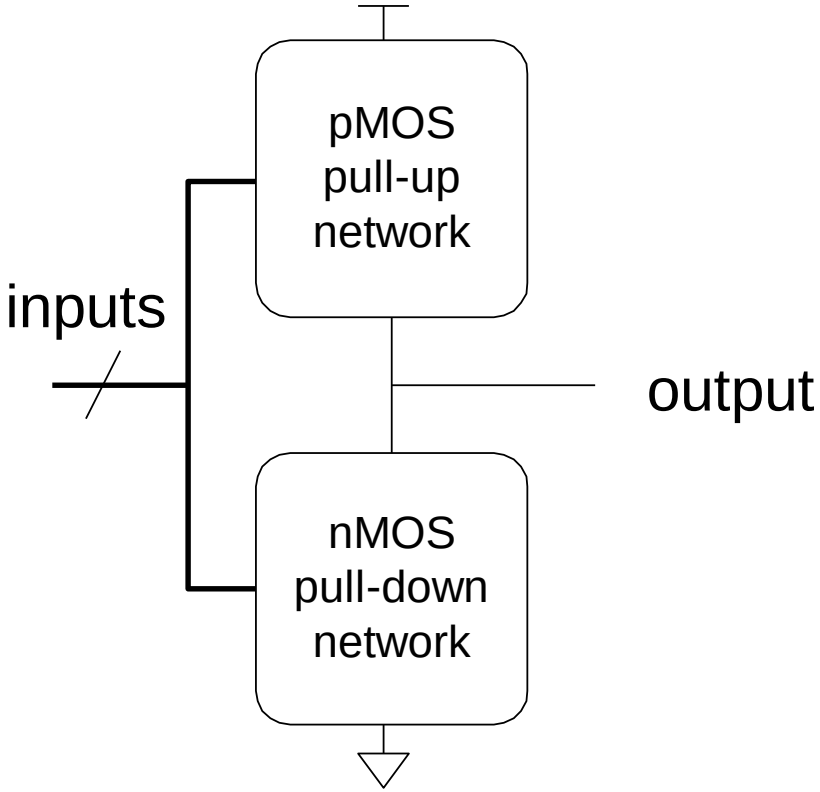
- The general form used to construct any inverting logic gate, such as: NOT, NAND, or NOR
  - The networks may consist of transistors in series or in parallel
  - When transistors are in parallel, the network is **ON** if one of the transistors is **ON**
  - When transistors are in series, the network is **ON** only if all transistors are **ON**



pMOS transistors are used for pull-u**p**
nMOS transistors are used for pull-dow**n**

General CMOS Gate Structure (II)

- Exactly one network should be ON, and the other network should be OFF at any given time
  - If both networks are ON at the same time, there is a short circuit → likely incorrect operation
  - If both networks are OFF at the same time, the output is floating → undefined



pMOS transistors are used for pull-u**p**
nMOS transistors are used for pull-dow**n**

We Covered Until Here in Lecture

Digital Design & Computer Arch.

Lecture 1: Introduction: Fundamentals, Transistors, Gates

Prof. Onur Mutlu

ETH Zürich

Spring 2025

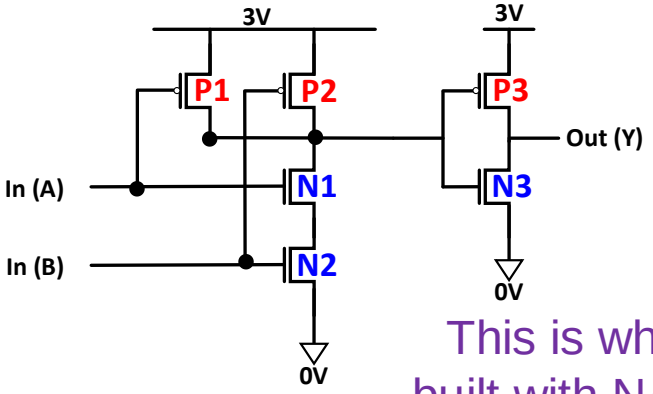
20 February 2025

Further Slides for Your Own Study (May Be Covered in Future Lectures)

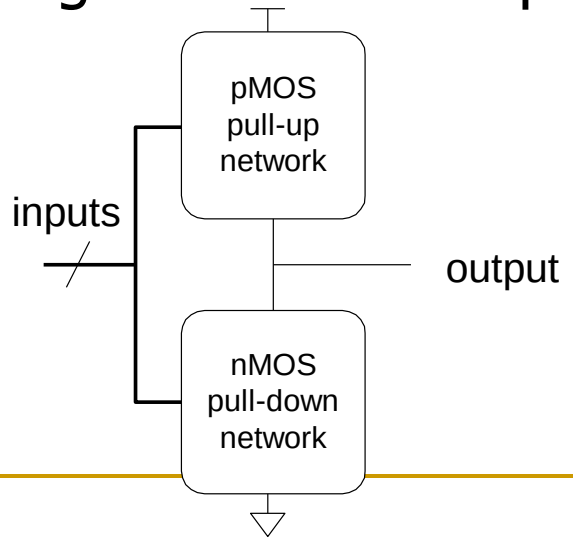
Digging Deeper: Why This Structure?

- MOS transistors are **imperfect** switches
- pMOS transistors pass 1's well but 0's poorly (holes carry charge)
- nMOS transistors pass 0's well but 1's poorly (electrons carry charge)

- **p**MOS transistors are good at "pulling u**p**" the output
- **n**MOS transistors are good at "pulling dow**n**" the output



This is why AND is built with NAND + NOT



See Section 1.7 in H&H

Digging Deeper: Latency

- Which one is slower?
  - Transistors in series
  - Transistors in parallel

- Series connections are slower than parallel connections
  - More resistance on the wire

- How do you alleviate this latency?
  - See H&H Section 1.7.8 for an example: pseudo-nMOS Logic

Used in the past when pMOS transistors could not be fabricated well

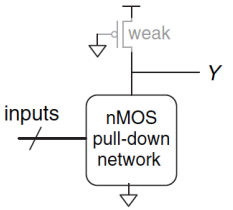


Figure 1.39 Generic pseudo-nMOS gate

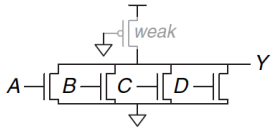


Figure 1.40 Pseudo-nMOS four-input NOR gate

Digging Deeper: Power Consumption

- **Dynamic Power Consumption**
  - Power used to charge capacitance as signals change (0 ←→ 1)
  - $C * V^2 * f$
    - C = capacitance of the circuit (wires and gates)
    - V = supply voltage
    - f = charging frequency of the capacitor

- **Static Power Consumption**
  - Power used when signals do not change
  - $V * I_{leakage}$
    - supply voltage * leakage current

- **Energy Consumption**
  - Power * Time

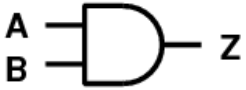
See more in H&H Chapter 1.8

Common Logic Gates

Buffer

A	Z
0	0
1	1

AND



A	B	Z
0	0	0
0	1	0
1	0	0
1	1	1

OR

A	B	Z
0	0	0
0	1	1
1	0	1
1	1	1

XOR

A	B	Z
0	0	0
0	1	1
1	0	1
1	1	0

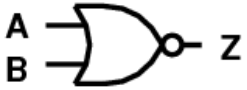
Inverter

A	Z
0	1
1	0

NAND

A	B	Z
0	0	1
0	1	1
1	0	1
1	1	0

NOR



A	B	Z
0	0	1
0	1	0
1	0	0
1	1	0

XNOR

A	B	Z
0	0	1
0	1	0
1	0	0
1	1	1

Larger Gates

- We can extend the gates to more than 2 inputs
- Example: 3-input AND gate, 10-input NOR gate
- See your readings

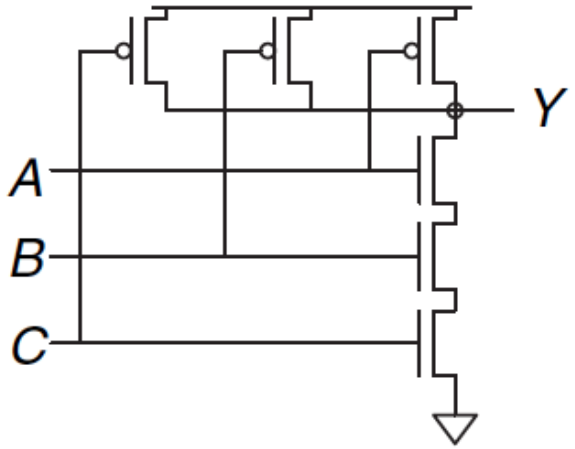
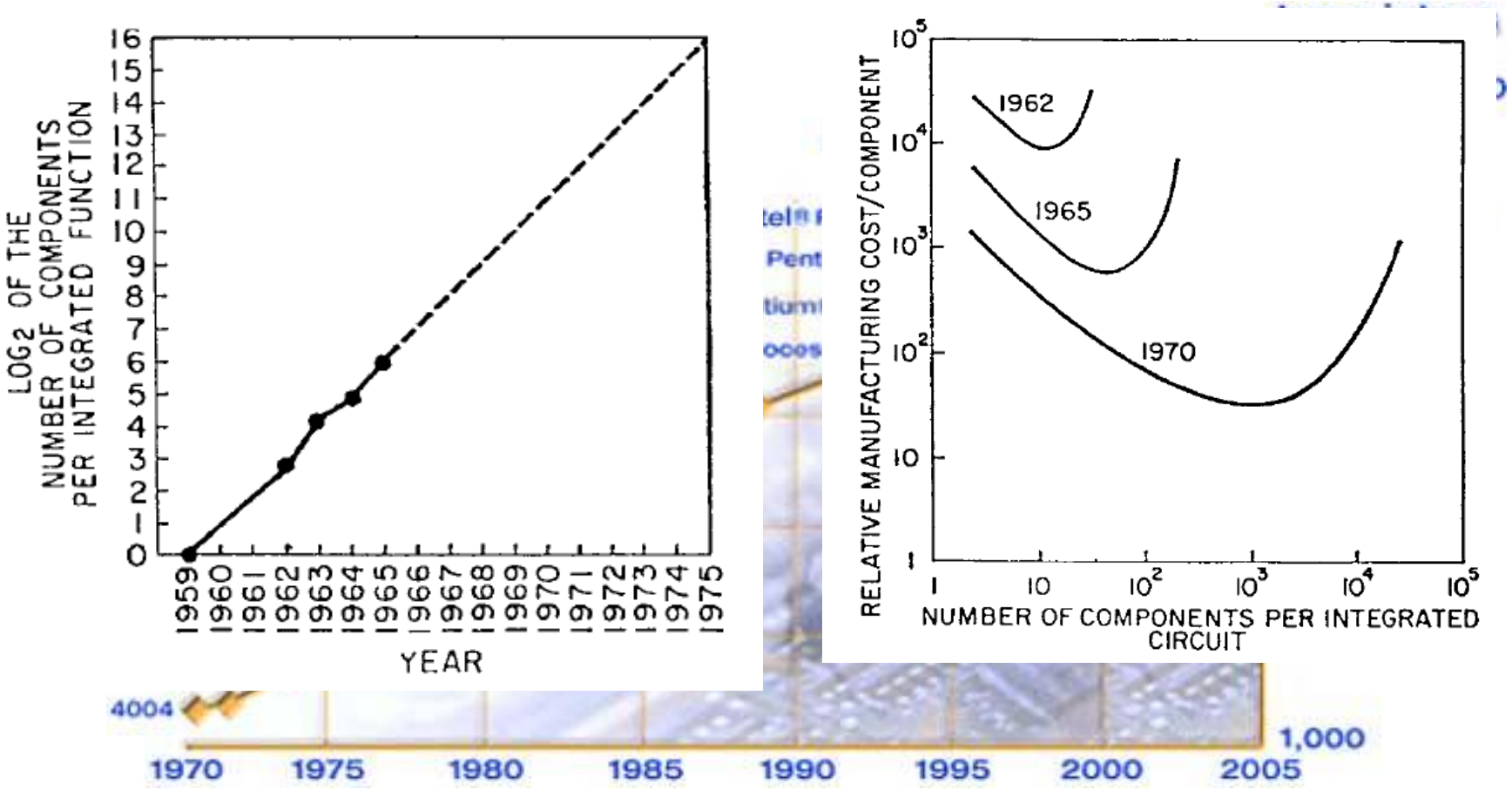


Figure 1.35 Three-input NAND gate schematic

Aside: Moore's Law: Enabler of Many Gates on a Chip

An Enabler: Moore's Law

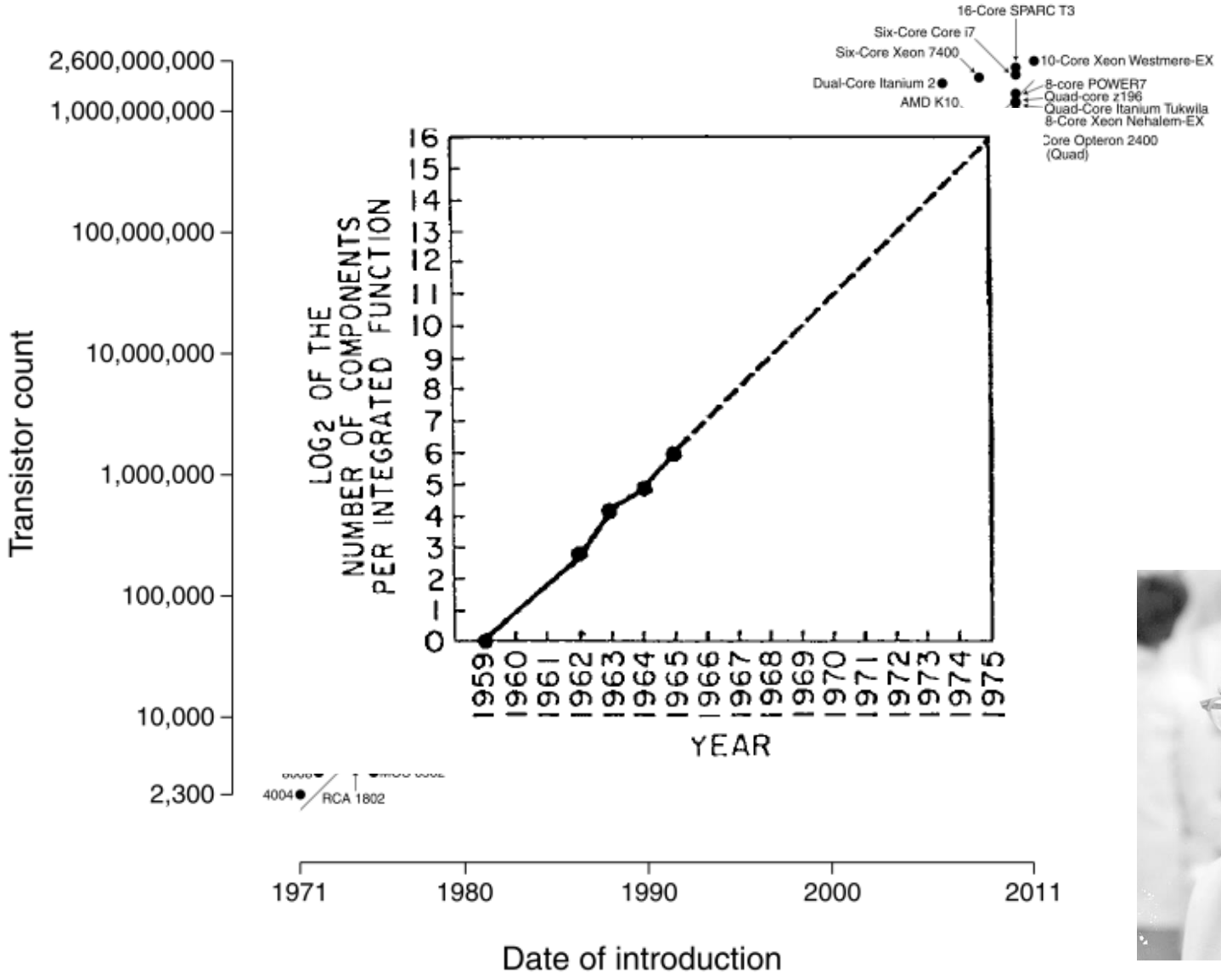


Moore, "Cramming more components onto integrated circuits," Electronics Magazine, 1965.

Component counts double every other year

Image source: Intel

Microprocessor Transistor Counts 1971-2011 & Moore's Law



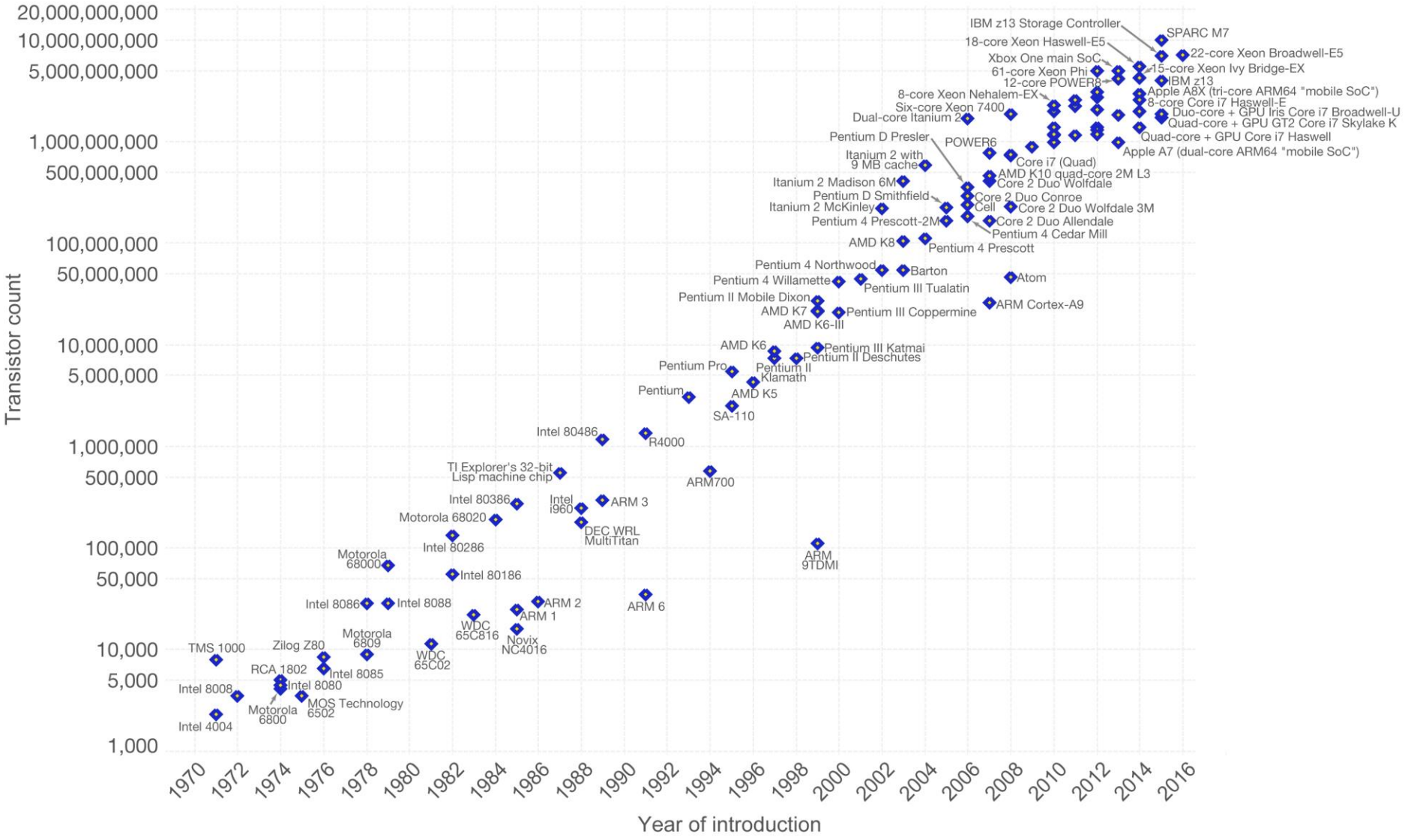
Number of transistors on an integrated circuit doubles ~ every two years

Image source: Wikipedia

Moore's Law – The number of transistors on integrated circuit chips (1971-2016)

Our World in Data

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important as other aspects of technological progress – such as processing speed or the price of electronic products – are strongly linked to Moore's law.



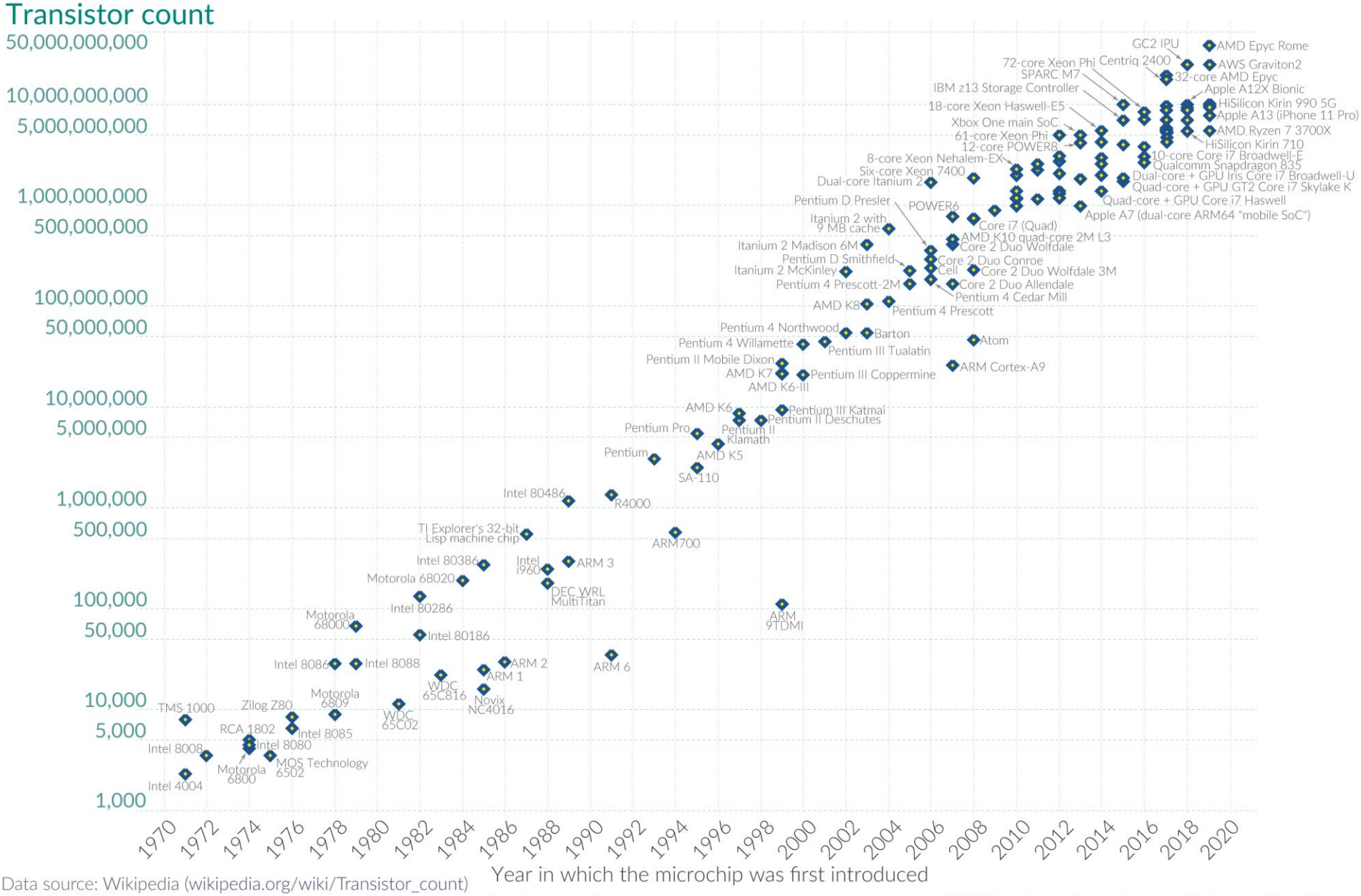
Data source: Wikipedia (https://en.wikipedia.org/wiki/Transistor_count)

The data visualization is available at OurWorldinData.org. There you find more visualizations and research on this topic.

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

Our World in Data

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.



Data source: Wikipedia (wikipedia.org/wiki/Transistor_count)

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

Recommended Reading

- Moore, "Cramming more components onto integrated circuits," Electronics Magazine, 1965.

- Only 3 pages

- A quote:

  "*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.*"

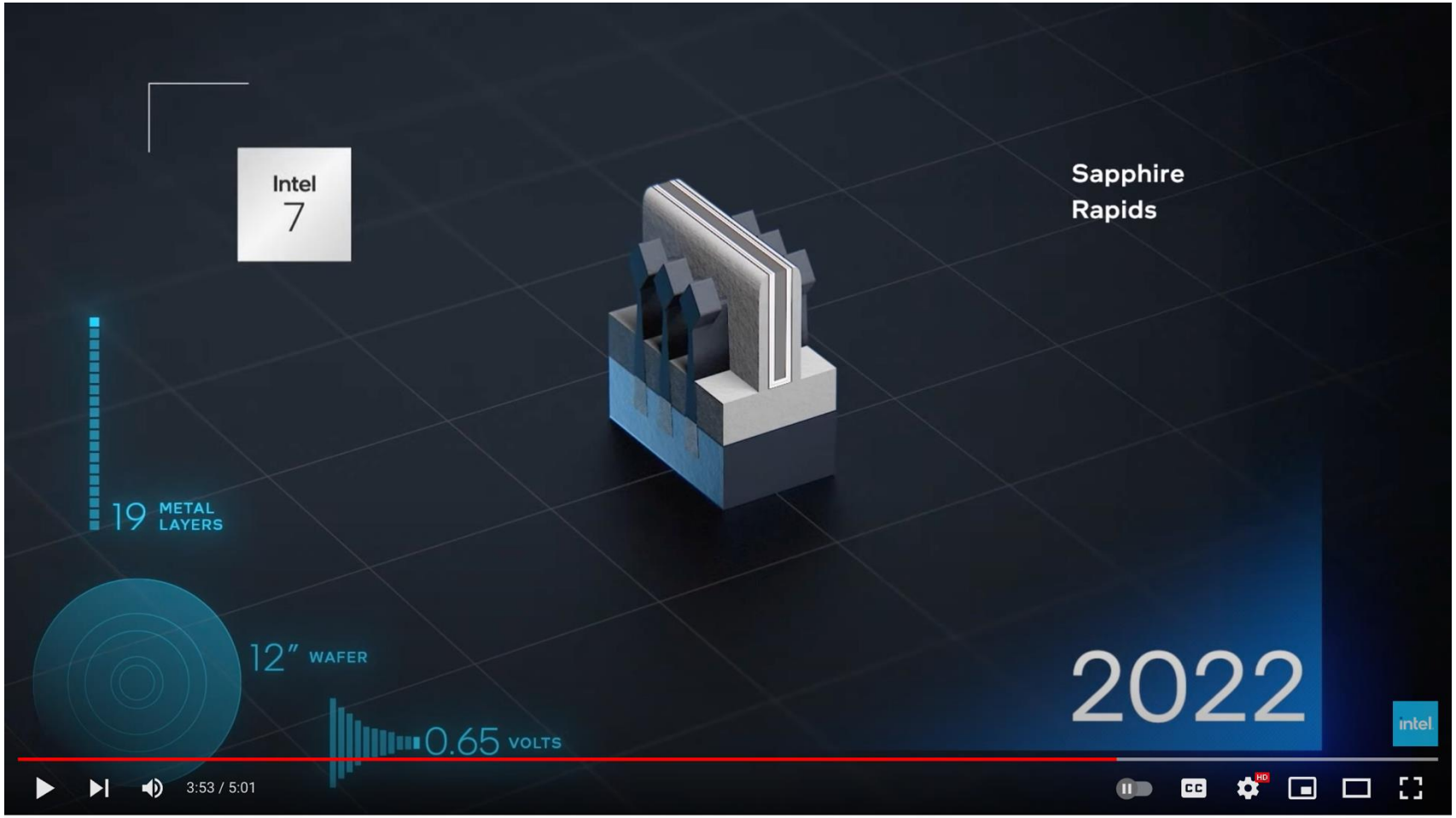
- Another quote:

  "Will it be possible to remove the heat generated by tens of thousands of components in a single silicon chip?"

How Do We Keep Moore's Law: Innovation

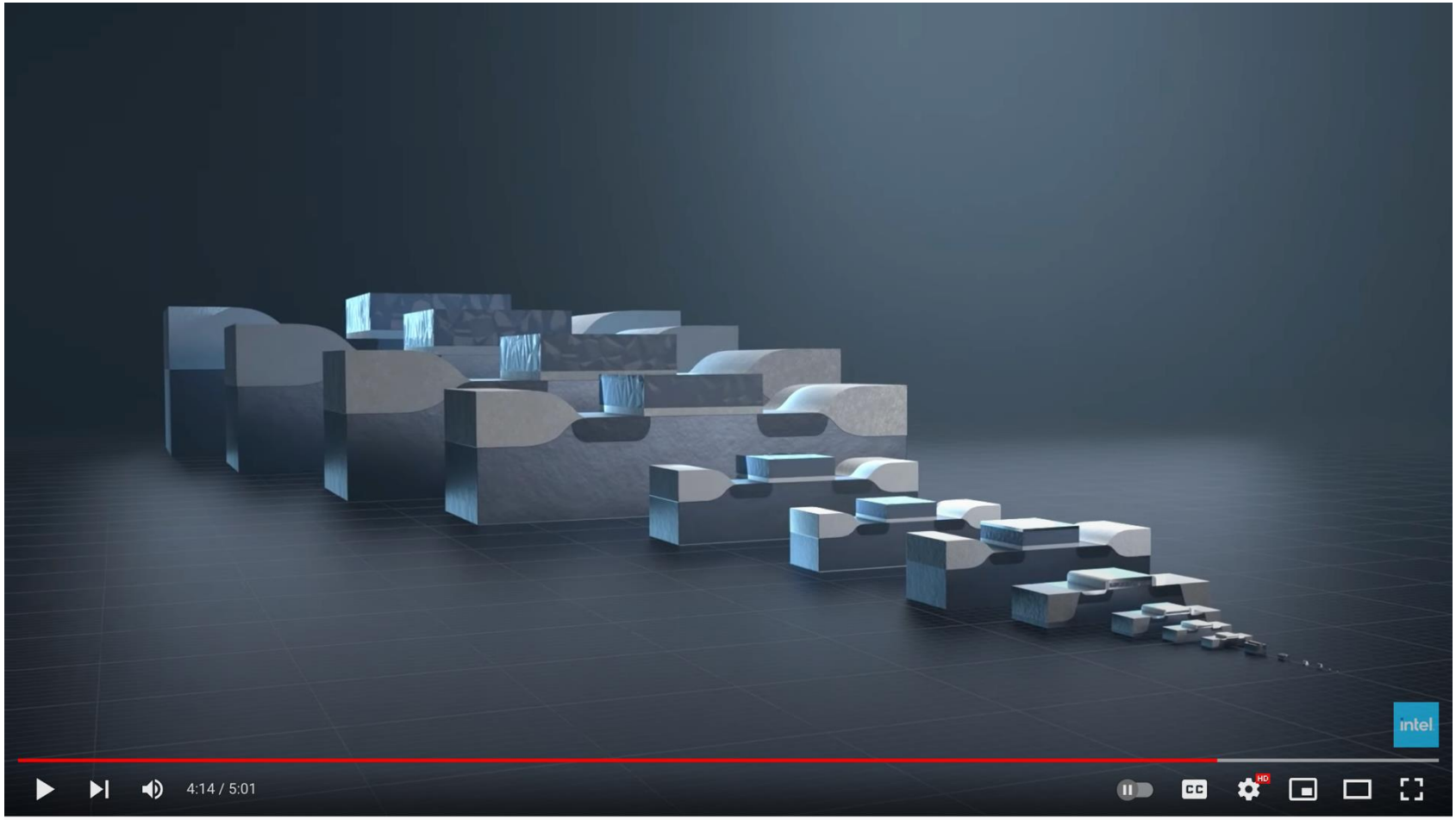
- **Manufacturing smaller transistors/structures**
  - Some structures are already a few atoms in size
- **Finding materials with better properties**
  - Copper instead of Aluminum (better conductor)
  - Hafnium Oxide, air for Insulators
  - Making sure all materials are compatible is the challenge
- **Enabling precision manufacturing**
  - Extreme ultraviolet (EUV) light to pattern <10nm structures
- **Creating new device technologies**
  - FinFET, Gate All Around transistor, Single Electron Transistor…

A 5-Minute Video on Transistor Innovation



https://www.youtube.com/watch?v=Z7M8etXUEUU

A 5-Minute Video on Transistor Innovation



Evolution of Transistor Innovation

12,460 views • Feb 22, 2022

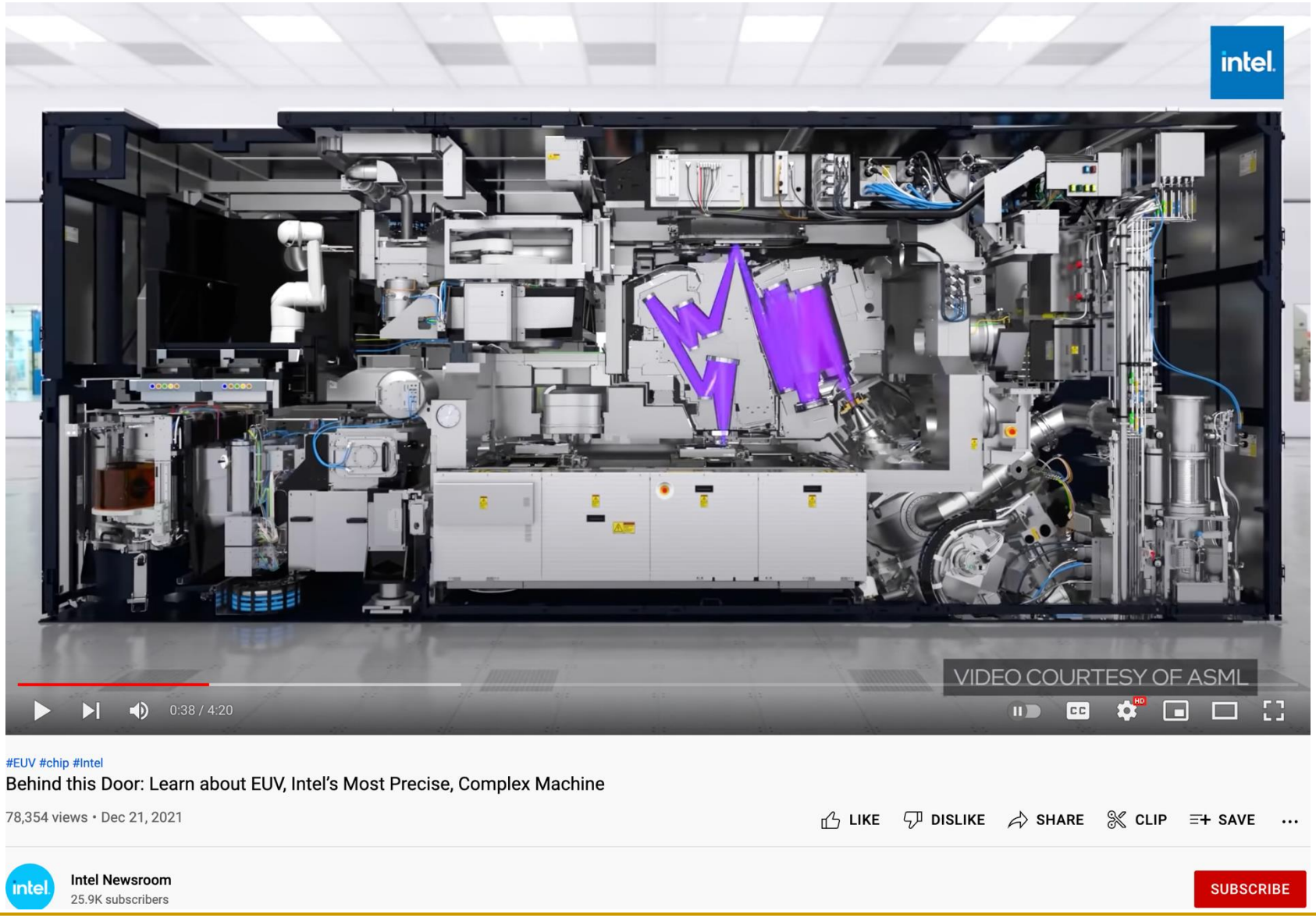
628 DISLIKE SHARE CLIP SAVE ...

intel Intel Technology
15.5K subscribers

SUBSCRIBE

https://www.youtube.com/watch?v=Z7M8etXUEUU

Enabling Manufacturing Tech: EUV



https://www.youtube.com/watch?v=Jv40Viz-KTc

Enabling Manufacturing Tech: EUV

Extreme ultraviolet lithography

11 languages

Article Talk Read Edit View history

From Wikipedia, the free encyclopedia

Extreme ultraviolet lithography (also known as **EUV** or **EUVL**) is an optical lithography technology used in semiconductor device fabrication to make integrated circuits (ICs). It uses extreme ultraviolet (EUV) wavelengths, roughly spanning a 2% FWHM bandwidth near *13.5 nm* (13.36nm - 13.65nm at 50% power), using a laser-pulsed *tin plasma*, to produce a pattern by using a reflective photomask to expose a substrate covered by photoresist. EUV (10-124nm) is the band between X-Rays (0.1-10nm) and overlapping slightly with UVC (100-280nm). It is currently applied only in the most advanced semiconductor device fabrication.

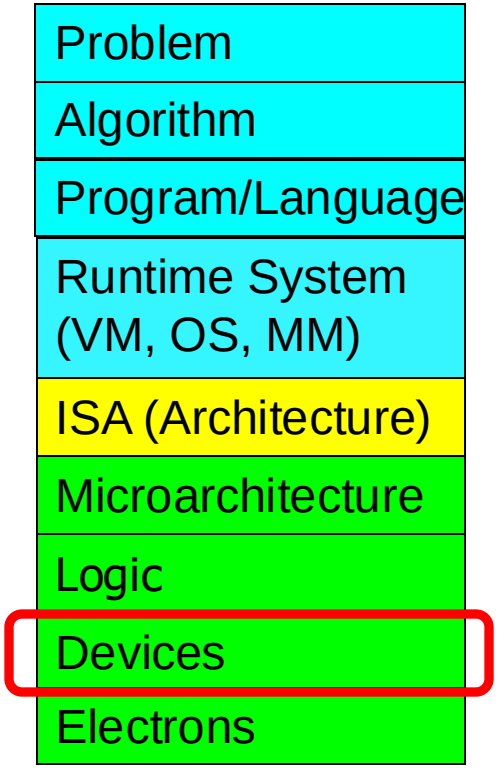
As of 2022, ASML Holding is the only company who produces and sells EUV systems for chip production, targeting 5 nm and 3 nm. At the 2019 International Electron Devices Meeting (IEDM), TSMC reported use of EUV for 5 nm in contact, via, metal line, and cut layers, where the cuts can be applied to fins, gates or metal lines.[1][2] At IEDM 2020, TSMC reported their 5 nm minimum metal pitch to be reduced 30% from that of 7 nm,[3] which was 40 nm.[4] Samsung's 5 nm is lithographically the same design rule as 7 nm, with a minimum metal pitch of 36 nm.[5]

History [edit]

In the 1960s, visible light was used for IC-production, with wavelengths as small as 435 nm (mercury "g line"). Later UV light was used, with wavelength of at first 365nm (mercury "i line"), then excimer wavelengths first of 248 nm (krypton fluoride laser) and then 193 nm (argon fluoride laser), which was called deep UV. The next step, going even smaller, was dubbed Extreme UV or EUV. The EUV technology was considered impossible by many. EUV is absorbed by glass and even air, so instead of using lenses, as before, to focus the beams of light, mirrors in a vacuum would be needed and a reliable production of EUV was also problematic. The then leading producers of steppers, Japanese companies Canon and Nikon gave up trying. And some even predicted the end of Moore's law.

https://en.wikipedia.org/wiki/Extreme_ultraviolet_lithography

Innovation At the Bottom Enables Computing



Historical: Opportunities at the Bottom

There's Plenty of Room at the Bottom

From Wikipedia, the free encyclopedia

"**There's Plenty of Room at the Bottom: An Invitation to Enter a New Field of Physics**" was a lecture given by physicist Richard Feynman at the annual American Physical Society meeting at Caltech on December 29, 1959.[1] Feynman considered the possibility of direct manipulation of individual atoms as a more powerful form of synthetic chemistry than those used at the time. Although versions of the talk were reprinted in a few popular magazines, it went largely unnoticed and did not inspire the conceptual beginnings of the field. Beginning in the 1980s, nanotechnology advocates cited it to establish the scientific credibility of their work.

Historical: Opportunities at the Bottom (II)

There's Plenty of Room at the Bottom

From Wikipedia, the free encyclopedia

Feynman considered some ramifications of a general ability to manipulate matter on an atomic scale. He was particularly interested in the possibilities of denser computer circuitry, and microscopes that could see things much smaller than is possible with scanning electron microscopes. These ideas were later realized by the use of the scanning tunneling microscope, the atomic force microscope and other examples of scanning probe microscopy and storage systems such as Millipede, created by researchers at IBM.

Feynman also suggested that it should be possible, in principle, to make nanoscale machines that "arrange the atoms the way we want", and do chemical synthesis by mechanical manipulation.

He also presented the possibility of "swallowing the doctor", an idea that he credited in the essay to his friend and graduate student Albert Hibbs. This concept involved building a tiny, swallowable surgical robot.

https://en.wikipedia.org/wiki/There%27s_Plenty_of_Room_at_the_Bottom

Extra Assignment 2: Moore's Law (I)

- **Paper review**
- G.E. Moore. "Cramming more components onto integrated circuits," Electronics magazine, 1965

- **Optional Assignment – for 0.5% extra credit**
  - **Write a 1-page individual review**
    - What are your key takeaways? What did you learn?
    - What surprised you about the content presented? What excited you?
    - How do you think Moore's law evolved since the paper was written?
    - What do you think (future) solutions should be like?
    - Submit your summary to Moodle – deadline March 21